Can Ontologies Systematically Help in the Design of Domain-Specific Visual Languages?

Maria das Graças da Silva Teixeira, Ricardo de Almeida Falbo, and Giancarlo Guizzardi

Ontology and Conceptual Modeling Research Group (NEMO), Federal University of Espírito Santo, Vitoria/ES, Brazil
maria.teixeira@ufes.br, {gguizzardi,falbo}@inf.ufes.br

Abstract. There has been a growing interest in Domain-Specific Visual Modeling Languages (DSVL) and their support for domain understanding and communication. However, the quality of these languages fundamentally depends on how well their structure reflects the structure of the abstractions constituting the underlying domain conceptualization. Since a well-founded domain ontology aims at faithfully representing a domain, it can be seen the ideal input for engineering a DSVL. In this paper, we present an experiment that analyses the performance of computer students in interpreting instance models by varying the concrete syntax of the language used. We contrast a generic notation (UML-based notation for object diagrams) and a domain specific notation that was designed based on a well-founded ontology for the domain of organizational structures. The hypothesis is that the performance of participants in interpreting the models using the domain specific notation is better than those who do it through a generic notation. Performance is evaluated by taking response time and correctness of the answers into account. The results confirm, but also contradict the hypothesis initially formulated.

Keywords: Domain-Specific Visual Language, Ontology, Empirical Study, Conceptual Modeling.

1 Introduction

Conceptual Modeling is a fundamental activity to many areas in Computer Science, such as Ontology Engineering, Knowledge Representation and Software Engineering. In his seminal paper [1], Mylopoulos defines Conceptual Modeling as the activity of formally describing in diagrammatic notation aspects of the physical and social world for the purposes of understanding, communication and problem-solving. In conceptual modeling, visual notations play a key role, especially in communicating with domain experts [2] and in supporting more effective understanding and recall of domain models [3].

Recently, there has been a growing interest in the design and use of Domain-Specific Visual Modeling Languages (DSVL). According to [4], a DSVL follows the domain abstractions and semantics, allowing developers to perceive themselves as

R. Meersman et al. (Eds.): OTM 2013, LNCS 8185, pp. 737–754, 2013.

working directly with domain concepts, leading to major improvements in productivity, time-to-market responsiveness and training time.

Gurr [5] argues that the stronger the match between a conceptualization of a domain and its representing model, the easier it is to reason with the latter. The interpretation of a diagram by a person correlates precisely and uniquely with the conceptualization being represented. A direct consequence of this fact is that the more we know about the subject domain being represented, the better we can systematically exploit its properties in the design of a DSVL for that domain. To put in language engineering terms, the quality of a DSVL depends fundamentally on the quality of the representation of the subject domain that is used as input. As argued by [3]: "*cognitive effectiveness is not an intrinsic property of visual representations but something that must be designed into them. […] There can be huge differences between effective and ineffective diagrams and ineffective diagrams can be less effective than text*".

Domain ontologies, in Computer and Information Science, are taken as explicit and formal representations of domain conceptualizations. In its original meaning, ontologies are supposed to be *reference models* of a given domain. In [6], Guizzardi proposed a language engineering framework that advocates the use of high-quality domain ontologies as the ideal representation for a subject domain and, hence, the ideal input conceptual model for the design of a DSVL. According to him, by making full use of a system of formal ontological meta-properties in the representation of the domain ontology at hand, the visual concrete syntax of a DSVL can be systematically designed to increase its cognitive and pragmatic efficiency. Thus, high-quality domain ontologies can produce better domain-specific languages, because they ensure a proper representation for their subject domain, making explicit a number of ontological meta-properties of domain concepts that can be employed in the design of a system of concrete syntax [6].

However, for a body of knowledge to be considered scientific, its truth and validity must be proven. A particular item of knowledge is considered to be scientifically valid if it has been checked against reality [7]. Thus, for evaluating the language engineering framework proposed in [6], we performed an empirical study, which is reported in this paper. The *goal* of this study is to compare a DSVL that was designed based on a well-founded ontology for the domain of organizational structures against a generic notation (UML-based notation for object diagrams). The *research hypothesis* is that the performance of the participants in interpreting instance models using the domain-specific notation is better than that made by participants interpreting instance models written in the generic notation. Performance is evaluated considering two measures: response time and correctness of the answers. The *subjects* are Computer Science students that have some experience in conceptual modeling. The results contain indications that the hypothesis is only partially valid, since we also obtained unexpected results, pointing to the need for further studies.

The remainder of this paper is organized as follows. Section 2 discusses the relationship between ontology and visual language concrete syntax. Section 3 presents a fragment of the ontology for organizational structures and discusses how this ontology and its meta-properties were exploited in the design of a DSVL. Section 4 presents the empirical study itself. Section 5 presents our final considerations.

2 Ontology and Visual Concrete Syntax

As discussed in [6], the appropriateness of an ontology can be engineered by guaranteeing an isomorphism between a concrete representation of a subject domain (a reference ontology for the domain) and a concrete representation of a language system (a language metamodel), i.e., the mapping between the ontological distinctions countenanced by the domain ontology and the modeling primitives constituting the language metamodel should be a one-to-one mapping. If this isomorphism is broken, we have cases such: (i) there are elements in the domain that cannot be expressed using the language, hurting language expressivity and domain appropriateness; (ii) there are elements in the domain that are expressed by more than one element in the language, introducing ambiguity and hurting clarity and comprehensibility appropriateness; (iii) there are elements in the language that don´t have an interpretation, hurting comprehensibility; and (iv) there are elements in the domain that are represented by more than one element in the language, hurting simplicity and interpretation.

In line with [6], this isomorphic mapping is defined only between ontology and abstract syntax (the modeling capabilities of the language). Moody [3] makes an analogous claim regarding the relation between concrete syntax and abstract syntax and, indirectly, the relationship between concrete syntax and ontology. This idea is encoded in the quality principle for visual notations that he terms ***semiotic clarity***. By discussing semiotic clarity, Moody draws from Nelson Goodman's *Theory of Symbols* when advocating *"for a notation to satisfy the requirements of a notational system, there should be a one-to-one correspondence between symbols and their referent concepts"* [7]. Here, once more, when the isomorphism is broken, the following anomalies occur [3]: (a) *Symbol redundancy* exists when multiple symbols are used to represent the same semantic construct; (b) *Symbol overload* exists when the same graphical symbol is used to represent different semantic constructs; (c) *Symbol excess* exists when graphical symbols are used but they do not represent any semantic construct; (d) *Symbol deficit* exists when semantic constructs are not represented by any graphical symbol.

Given that the suitability of a visual notation is evaluated with respect to a domain ontology, semiotic clarity of a system of visual syntax essentially depends on the characteristics of the underlying domain represented in that domain ontology. One aspect, however, which is not evident in the framework proposed by Moody, is the following. Unlike in the case of textual syntaxes, the graphical symbols that form the system of concrete syntax often fall naturally into a hierarchical typing, which informs about the semantics of what is being represented. An analogous statement can be made regarding certain relations between graphical symbols (e.g., spatial relations) that can be systematically mapped onto semantic relations with equivalent logical properties. These features of graphical symbol systems and relations are illustrated by the example in Fig. 1.

As one can notice, the models of Fig. 1.(a) and 1.(b) are isomorphic. The different concrete kinds of entities in the model (Federal Capital, State, Metropolis and Town) are represented by different kinds of geometrical objects (Square, Non-Squared Rectangle, Large Circle and Small Circle). In particular, the taxonomic structure of

Geopolitical Units is isomorphic to the one of Geometric Figures. For this reason, in a visual query one can immediately notice that Federal Capital is more similar to a State than to a City, and probably shares a common super-type with the former. Notice that, if we had produced a different taxonomic structure in the model of Fig. 1.(a), then a different choice of representing graphical symbols would have been made, possibly creating undesired implicatures for the model reader [6]. Given the difficulties experienced by modelers in the design of domain taxonomic structures [8], this illustrates the importance of having a well-designed ontology for the design of semiotic clarity in the system of visual syntax.

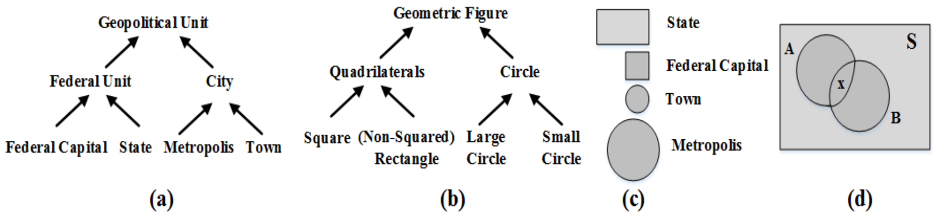


Fig. 1. (a) A fragment of a taxonomy for the geopolitical domain; (b) a taxonomy of geometric objects isomorphic to the structure in (a); (c) a system of visual symbols from (b) to represent the domain concepts in (a); (d) a representation of parthood relations between a state S, the cities A and B and one of its common part x

There is an additional point worth mentioning about this example. Although City and Metropolis/Town are examples of classifiers (classes, types), they classify their instances in very different ways with respect to modality. City is what is termed *essential* classifier, i.e., it *classifies* its instances necessarily (in the modal sense). In contrast, Metropolis/Town is a *contingent* classifier and classifies its instances only *accidently* (once more, in the modal sense). In other words, while instances of city are always instances of city and cannot cease to be so without ceasing to exist, the same city can be considered a town in a world w and a metropolis in w' while still maintaining its cross-world identity (cities change from town to metropolis given the size of their population). Likewise, in Fig. 1.(b), the color property of a geometric figure is considered one of its contingent intrinsic properties. Thus, a particular circular form is assumed to be able to change its color while maintaining a continuous visual percept. Furthermore, the intrinsic property population size that motivates the change in classification of cities is associated with a linearly ordered dimension. For this reason, we have decided to associate the intrinsic property of circles that represent this classifier variation by employing also a linearly ordered dimension (size).

In Fig. 1.(d), we have a representation of two cities A and B, which are part of a state S. By looking at this model, one can immediately infer that x, which is a part of city A (and B) is also a part of S (let us suppose x a neighborhood which is jointly managed by the two cities) . Moreover, one can immediately infer that x is a shared part of cities A and B. These inferences are termed *inferential free rides* [9], as they are considered to be costless from a cognitive point of view. These inferential free rides are present in this model because of the relation used to represent parthood, namely, the spatial inclusion in the plane. For instance, it is because spatial inclusion is a transitive relation that we can automatically derive (by transitivity) that x is part of S [6].

Although authors such as Moody and Gurr explicitly acknowledge the influence of suitable domain representations for the design of modeling languages, they do not properly elaborate on how these high-quality domain representations should be produced. Gurr simply alludes to the representation of domain models as algebraic structures, without offering any methodological and engineering support for their construction. Moody explicitly acknowledges the role of ontologies in the representation of semantic domains, but offers no discussion what so ever on how these semantic domain representations are created.

In a number of works [10][11], we have demonstrated the importance of foundational theories (i.e., domain-independent formal ontological theories) for analyzing and (re)designing domain reference ontologies. In particular, in the approach presented in [6], a philosophically and cognitively well-founded ontology representation language is used for creating domain ontologies, which, in turn, are used as input for designing domain-specific visual languages. This ontology representation language (termed OntoUML [12]) incorporates as modeling primitives, a number of formal meta-properties put forth by the Unified Foundational Ontology (UFO). UFO is a foundational theory incorporating a number of results from formal ontology in philosophy, philosophical logic, linguistics and cognitive science. For a full account of UFO, its formal characterization and empirical support, one should refer to [12][13].

A direct contribution of OntoUML to the design of *Domain-Specific Languages* is the following. Given that OntoUML incorporates in its metamodel the axiomatization of UFO, the only grammatically correct domain models that can be produced in this language are ontologically consistent ones, i.e., domain models that are consistent with UFO's basic axiomatization. Another contribution, however, is that, contrary to the merely formal structures employed in [5], the domain ontologies represented in OntoUML capture a number of ontological meta-properties that are used to further qualify the ontological status of domain concepts. We concentrate here on a fragment of OntoUML with a focus on *Object Types* and *Part-Whole* relations.

A fundamental modal meta-property used to distinguish among categories of *Object Types* is *rigidity* (and the associated notion of *anti-rigidity*). Formally, we have that [8]: a type T is rigid iff every instance of T is necessarily an instance of T (in the modal sense). In contrast, a type T' is anti-rigid iff for every instance x of T' there is a possible situation in which x is not an instance of T'. A stereotypical example that illustrates this distinction is the types Person and Student: instances of Person are necessarily so (Person is a rigid type); in opposition, instances of Student are merely contingently so (Student is an anti-rigid type).

Kinds and ***Subkinds*** are object types that are rigid [8]. These types define a taxonomy of rigid types instantiated by a given individual (kind being the unique topmost rigid type instantiated). Within the category of anti-rigid object types, we have a further distinction between ***Phases*** and ***Roles*** [8]. Both are specializations of rigid types. However, they are differentiated with respect to their *specialization conditions*. On Phases, the specialization condition is always an intrinsic one. For instance, a child is a Person whose age is in a certain range. On Roles their specialization condition is a relational one. For instance, a Student is a Person enrolled in a school.

A modal meta-property used to distinguish among the categories of Part-Whole relations is *existential dependence* [14]. An entity x is existentially dependent on another entity y iff in every situation that x exists then y must exist. Associated to existential dependence we have the notion of *generic dependence*. An entity x is generically dependent on a type Y iff in every situation where x exists an instance of Y must exist. These notions are used in UFO (among other things) to distinguish between part-whole relations that imply existential dependence and those that only imply generic dependence. A part-whole relation which implies only generic dependence from part to whole is named ***Parthood with Mandatory Wholes*** [14]. A part-whole relation which implies existential dependence from part to whole is termed ***Inseparable Parthood*** [14]. Another remark regarding part-whole relations worth mentioning is that contrary to purely formal mereological relations, part-whole relations which appear in conceptual models and material domain ontologies are non-transitive, i.e., they are transitive in certain situations and intransitive in others [15].

Finally, part-whole relations can be distinguished according to a meta-property named *shareability*. This meta-property wrongly defined in original UML specification has been refined in [12] with the following definition: (a) a (whole) type X is characterized by an exclusive (non-shareable) parthood relation with a (part) type Y iff every instance of X must have exactly one instance of Y as part; (b) a type X is characterized by a shareable parthood relation with a type Y iff instances of X can have more than one instance of Y as part.

3 From an Ontology of Organizational Structures to a Domain-Specific Visual Language for This Domain

Fig. 2 presents a small ontology of Organizational Structures. In this ontology, Employee is a role played by a Person when it is member of a Department. A Person (an abstract type) is either Man or Woman. An Employee is part of exactly one Department (represented by the non-shareable association end). Since this is a generic dependence relation, employees can change to different departments. An Employee is subordinated to at least one other employee who is its superior. Then, the types Subordinate Employee and Superior Employee are roles played by employees. As roles, an instance of Subordinate Employee can cease to be one, and for it to instantiate this role, there must exist another Employee instantiating the Superior Employee role. The same instance of Employee can simultaneously instantiate both roles.

A Department is part of exactly one Organizational Branch. Again, we have a case of a non-shareable parthood relation, but also one which implies existential dependency from part to whole (represented by the {inseparable} tag value), i.e., the Sales Department of an Organizational Branch can only exist as part of that branch. The relations between Employee and Department, and Department and Organizational Branch are cases of transitive parthood as identified in [15]. Commissions are collectives that have particular Employees as members (termed Commission Members).

Commissions can be in two different phases depending on the value of one of its intrinsic property (its amount of committed work). A Work-Overloaded Commission is a Commission such that its amount of committed work surpasses a certain threshold. A Normal Workload Commission is the complement of Commission with respect to Work-Overloaded Commission.

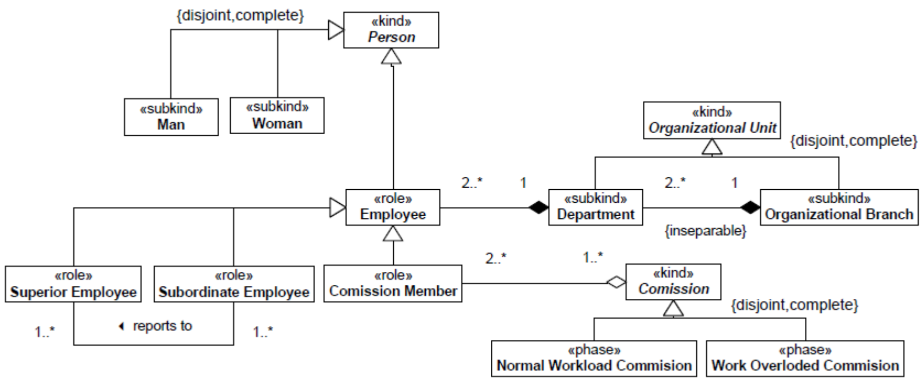


Fig. 2. A fragment of an ontology for organizational structures

Based on the ontology of Fig. 2, we have designed a domain-specific visual language aimed at representing valid instances of this ontology. This concrete syntax is the complement of the abstract syntax presented above, ideally being generated with the best that each of the theories presented in the previous section has to offer.

Table 1 presents the modeling primitives of this language via their respective concrete syntax. The table also relates these primitives with the domain concept they represent and with the ontological category of these domain concepts.

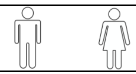
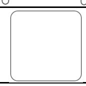
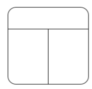
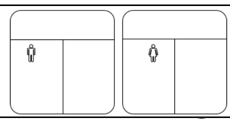
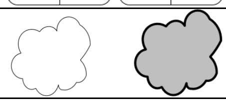
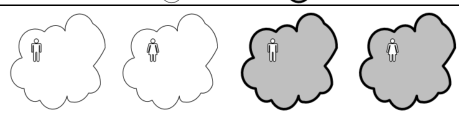
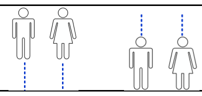
This concrete syntax presents semiotic clarity, i.e., there is an isomorphic mapping between the concepts in the domain ontology and the modeling primitives in the language. Moreover, the mapping between domain elements and elements in the visual notation takes full account of ontological categories and meta-properties of the former. Next, we elaborate on the systematic use of each of these ontological categories to derive properties of this system of concrete syntax.

Kinds and ***Subkinds***: In Fig. 2, we have both kinds and subkinds. As discussed in [16], shapes defined by closed contour are among the most basic metaphorical representations for objects. This idea is in line with a number of findings in cognitive science, including the one that shape plays a fundamental role in kind classification [17]. In the language defined in Table 1, each concrete subkind is associated with a shape. The chosen shapes are sufficiently dissimilar and are aligned with the taxonomic relations between domain types as presented in Fig. 2. For instance, the "four-sized" figures used to represent Organizational Branches and Departments are similar, considering they are organizational units. On the other hand, they are dissimilar from the blobs used to represent Commissions. These features highlight the characteristic of ***perceptual discriminability*** pointed by Moody and Hillegersberg [18].

Another aspect is the direct metaphorical resemblance between the graphical elements used and their referents. A case is the iconic representation for *Man* and

Woman. The representation of *Departments* as "pieces of an *Organizational Branch*" is adherent to the idea of "organizational divisions" associated to *Departments*. In addition, while the straight lines used in the contour of *Organizational Units* seems a more formal and rigid structure, the round boundaries of blobs representing *Commissions* are more naturally associated with a flexible informal one. The systematic use of these metaphorical resemblances brings to this notational system another important quality characteristic according to Moody, namely, ***perceptual immediacy*** [18].

Table 1. Visual concrete syntax for the organization structure ontology of Fig. 2

Domain Type	Ontological Category	Notational Element
Person, Organizational Unit, Commission	Kind	Abstract class; No direct representation
Man, Woman	Subkind	
Organizational Branch	Subkind	
Department and Department *is component of* Organizational Branch	Subkind and part-whole relation	
Employee and Employee *is component of* Department	Role and part-whole relation	
Normal Load Commission, Overloaded Commission	Phase	
Commission Member and Commission Member is part of Commission	Role and part-whole relation	
Superior and Subordinate Employees	Role	
Subordinate Employee *reports to* Superior Employee	Domain association	Combination of *is-dashed-line-connected* with the *above* relation in the plane

Phases: We used an intrinsic property of visual percept to represent different phases of a kind (the entity can change its phase but maintain its identity). In the language presented in Table 1, the changes in color of blobs used to represent *Commissions* represent different phases. We use a high-saturation color to represent the *Work-Overloaded Commission* exploring a metaphorical relation between "more quantity of color" and "more quantity of work". This feature increases its ***perceptual immediacy***. The difference in brightness of grey hue used to represent an overloaded commission and white one used to represent a regular load commission creates an efficient ***perceptual pop-out*** [18]. Finally, given that identifying overload commissions is an important task in the domain, the perceptual pop-out is increased by the increased ***perceptual discriminability*** between these two phases. This is due to the use of a different thickness of blobs boundaries. This is a case of ***redundant coding*** in [18].

<u>Relations</u>: In the ontology of Fig. 2, there are parthood relations between: (1) *Employee* and *Department*, and (2) *Department* and *Organizational Branch*. They are irreflexive and asymmetric. Moreover, transitivity holds across (1) and (2). By using the relation of spatial inclusion in the plane to represent these relations, we have a mapping to a visual relation that has exactly the same formal properties of the represented one, since spatial inclusion is also a partial order relation.

The different *Departments* that comprise an *Organizational Branch* are represented by a tessellation of the spatial region used to represent that branch. The lack of overlap between these regions allows for a ***perceptually immediate*** representation of the non-shareability metaproperty of these relations. This representation also contributes to ***perceptual immediacy*** due to the fact that, if *Departments* are represented as partitions of the region representing its associated *Organizational Branches*, this also favors the interpretation of existential dependence from part to whole.

Another parthood relation is between *Commission Member* and *Commission*. This relation is represented as a spatial containment relation between icons representing *Person* and a blob representing *Commission*. These blob forms can overlap with *Department* regions. This feature allows for direct inferential free ride on the identification of which *Department* a *Commission Member* belongs to. In addition, in line with the ***shareability*** metaproperty of this relation in the ontology, one can easily imagine overlapping blobs allowing for a certain member to be part of multiple commissions.

A third relation is the *reportsTo* relation, defined between a *Superior Employee* and its *Subordinates*. We used a combination of visual relations to represent this association (we combined the above relation in the plane with the transitive closure of the is-dashed-line-connected relation). Additionally, the different texture of this line increases the ***perceptual discriminability*** when contrasting it to the solid lines used to demarcate *Department* partitions. Finally, the spatial metaphor of using "higher in the plane" to represent "higher in the hierarchy" favors ***perceptual immediacy***.

<u>Roles</u> and ***<u>Relational Properties</u>***: Finally, we need visual representations able to highlight roles and their relational properties. The roles *Employee* and Commission Member are represented by the contained in the region relation between a person icon and a region representing a *Department* and a *Commission*, respectively The roles *Supervisor* and *Supervised by* are represented by a dotted line between two person icons and their spatial positioning.

4 The Empirical Study

In this section, we describe the empirical study performed. The *goal* of the experiment is to collect indications about the use of the concrete syntax of the domain-specific language presented previously. The *research hypothesis* is that the performance of participants in interpreting instance models using the domain-specific notation is better (according to response time and correctness of answers) than that made by participants interpreting instance models written in a generic notation. The empirical study was conducted following the guidelines presented in [19].

The experiment has qualitative and quantitative *strategies*. The experimentation *level* is in-vitro (it was conducted in a controlled environment). The research *approach* is primarily analytical, to collect early indications for further experiments. The experiment has as its *object of study* two instantiations of the conceptual model presented in Fig. 2. The instantiations were presented in two different notations, a domain-specific notation and a generic notation, giving rise to four instance models.

The *subjects* are Computer Science students, from both under-graduate and post graduate levels, which attend classes of a Conceptual Modeling course. The minimum requirement expected for participating in the experiment was having basic knowledge of UML. A questionnaire was applied to capture the participants' profile. Regarding the sample size, there were 22 participants. They were divided into two groups (A and B) randomly. Group A has 12 participants, and Group B has 10 participants. The imbalance in number of participants occurred, because, until the draw, we had 23 participants, and Group B (which had 11 participants) had one participant less for the effective execution of the activity.

The *factor* of the experiment is the concrete syntax of a visual language, and the *alternatives* are: a generic notation (UML-based notation for object diagrams) and a domain-specific notation (presented in Table 1). The *task* is the interpretation of instance models for the same instantiations, using different notations. Questions regarding two instantiations of the conceptual model presented in Fig. 2 were posed, varying the concrete syntax of the language used for representing them. The first instantiation represented using the domain-specific notation is depicted in Fig. 3. A semantically equivalent representation using the generic notation is shown in Fig. 4. Each participant had to answer two questions about this instantiation: one subjective (Q1), and other objective (Q2). Another instantiation, similar to the first one but bigger, was also used and other two questions (Q3 and Q4) were posed. Q3 is subjective, while Q4 is objective. The questions and the predefined answers (separated by fragments) about the first instantiation are presented next. Regarding the second instantiation, the Q3 has 6 answer fragments, and the Q4 has 4 answer fragments.

1. *Consider the individual Lisa. What information can be obtained about this individual from the observation of the model?* Template Answer: Lisa is part of the Marketing Department (fragment 1). She is a woman (fragment 2). She is supervised by Mary (fragment 3). She supervises Ana (fragment 4) and Otto (fragment5). She is member of the Quality Control Commission (fragment 6).
2. *Which is(are) the employee(s) with the largest number of direct subordinates*? *How many are the subordinates of this(these*) *employee(s)*? Template Answer: Peter (fragment 1) and Robert (fragment 2). They have three direct subordinates each (fragment 3).

The dependent variables are: response time and correctness of the answers. These variables are measured for each question. The way to analyze response time is trivial:

the time taken to answer each question is recorded and the smaller it is the time, the best is the related notation. Correctness is measured by comparing the fragments of the participants' answers to the corresponding fragments of the template. If they are the same, then the fragment is *correct*; otherwise the fragment is *wrong*, if the participant said something wrong about the fragment, or *missing*, if the fragment in the template is not reported by the participant. This is what we call *fragment correctness*. We have also other two types of fragments: wrong complementary fragment, which occurs when the participant included in her answer a fragment that the model does not say, related or not to the question; and extra complementary fragment, when the participant included in her answer a fragment that is not part of the template, but it can be inferred from the model. An extra complementary fragment is not a mistake, and therefore it is simply ignored.

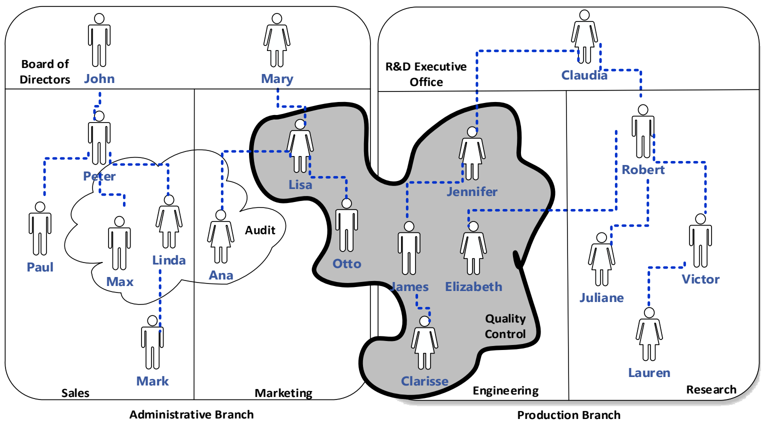


Fig. 3. An instance-model in the domain-specific language

To illustrate how we analyzed fragment correctness, consider the following answer given by a participant to question Q1: *"Lisa is part of the Board of Directors Department, and she is supervised by Mary. She is also part of the Marketing Department and she supervises Ana and Otto. She is part of the Quality Control Commission"*. According to the template, there are five correct fragments (1, 3, 4, 5 and 6), one missing (fragment 2), and one wrong complement (*Lisa is part of the Board of Directors Department*).

Besides interpreting the models, the participants filled in a questionnaire containing the following questions: (1) What is your impression (which one was easy/difficult, better/worse) among the models/concrete syntaxes presented? (2) Add any additional observation that you deem necessary.

Each group answered the same questions by interpreting each instantiation in a different notation. Group A answered the two questions (1 and 2) of the first instantiation in the domain-specific notation, while Group B answered the same questions of this instantiation using the generic notation. In the second instantiation, the situation was reversed, for answering questions 3 and 4, Group A interpreted the model written in the generic notation, while Group B interpreted the model written in the domain-specific notation.

In order to facilitate data collection, a website was developed. The site contained the instance models, the corresponding questions, and a link to the notation used in each instance. We recorded the participants' answer and the response time for each question automatically. Although we used a website, the experiment was conducted during a class in a lab, in order to ensure a stable Internet connection and to avoid distractions to participants, thereby reducing threats to the experiment.

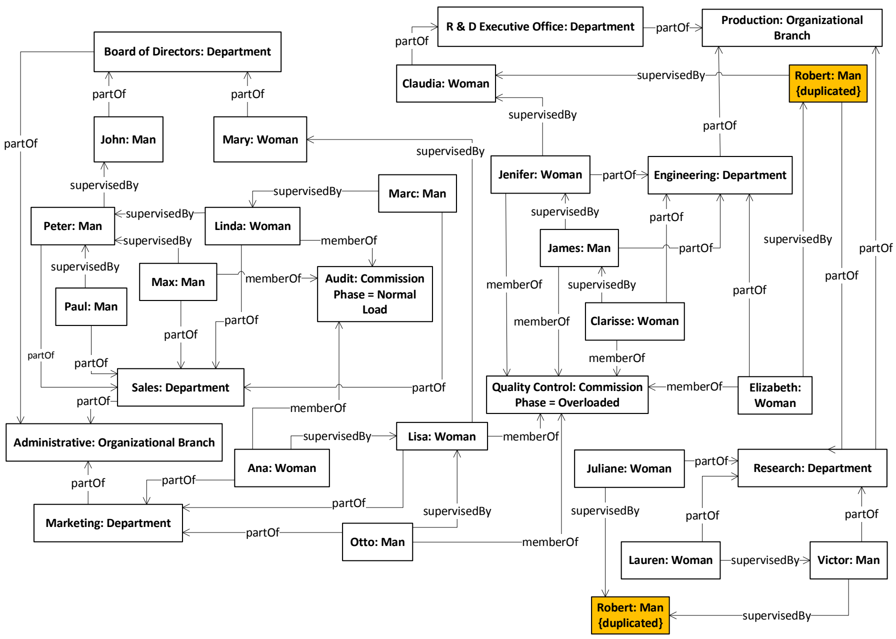


Fig. 4. An instance-model in the generic language

4.1 Collected Data

Regarding the participants' profile, we can say that: (i) The educational level (undergraduate, master and doctoral students) of the two groups were balanced; (ii) Regarding experience time in conceptual modeling, Group A has around 90% of its members with experience above 1 year, while Group B has 80% of participants in this range. We consider that the groups were balanced, even if members of Group A have a little more experience. The participants' profile was of students acting as modelers with some level of knowledge in conceptual modeling, specifically on using UML.

Table 2 presents data regarding the response time for each question. The columns present data on average, median, highest and lowest value of response time, and the percentage difference between highest and lowest averages for a question. Table 3 shows the percentage of fragment correctness for each question, and also the number of wrong complementary fragments, grouped by notation. In tables 2 and 3 we highlighted in grey the items that are consistent with our hypothesis, and in black the ones that contradict our hypothesis. Fig. 5 presents four graphs showing the response times for each question, comparing the notations. The values are ordered. Fig. 6 shows eight graphs showing the number of correct, wrong, missing, and wrong complementary fragments, two per question, comparing the notations. The values are ordered by number of correct fragments.

Table 2. Response Time (in seconds)

Question	Average (av)		Median		Highest Value		Lowest Value		(Smallest av / Largest av)
	Gr. A	Gr. B	Gr. A	Gr. B	Gr. A	Gr. B	Gr. A	Gr. B	
Q1	363,25	301,67	346,5	292	715	463	148	141	83,05%
Q2	101,92	210,22	99	226	157	324	49	79	48,48%
Q3	400,67	271,67	319,5	260	1416	453	153	117	67,80%
Q4	210,50	62,67	206,5	61	512	99	83	35	29,77%

Table 3. Percentage of correct, wrong and missing fragments, and number of wrong complementary fragments by notation

Question	% of Correct Fragments		% of Wrong Fragments		% of Missing Fragments		Number of Wrong Complementary Fragments	
	Specific	Generic	Specific	Generic	Specific	Generic	Specific	Generic
Q1	66,67%	85,00%	0,00%	3,33%	33,33%	11,671%	4	2
Q2	86,11%	90,0%	11,11%	3,33%	2,78%	6,67%	0	0
Q3	71,67%	88,89%	0,00%	0,00%	28,33%	11,11%	11	3
Q4	97,50%	75,00%	2,50%	12,50%	0,00%	12,50%	0	5

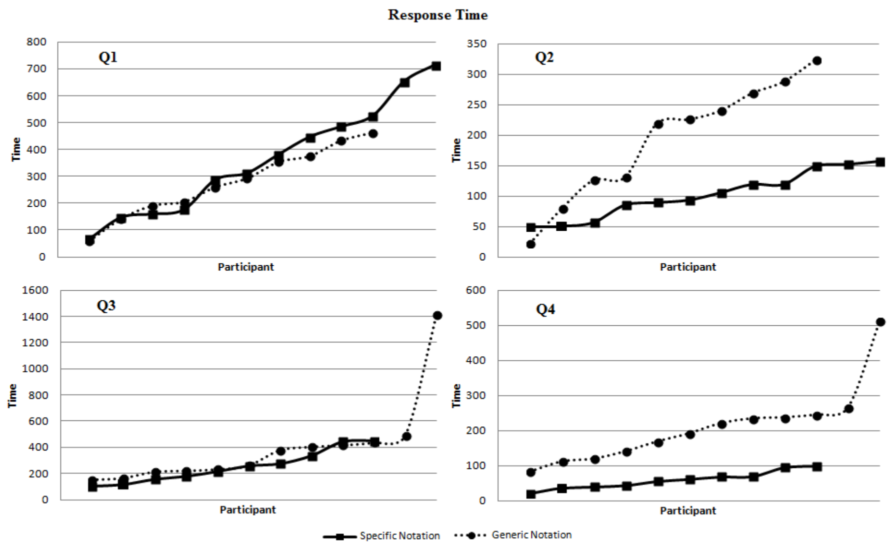


Fig. 5. Evolution of Response Time per Question for each Notation

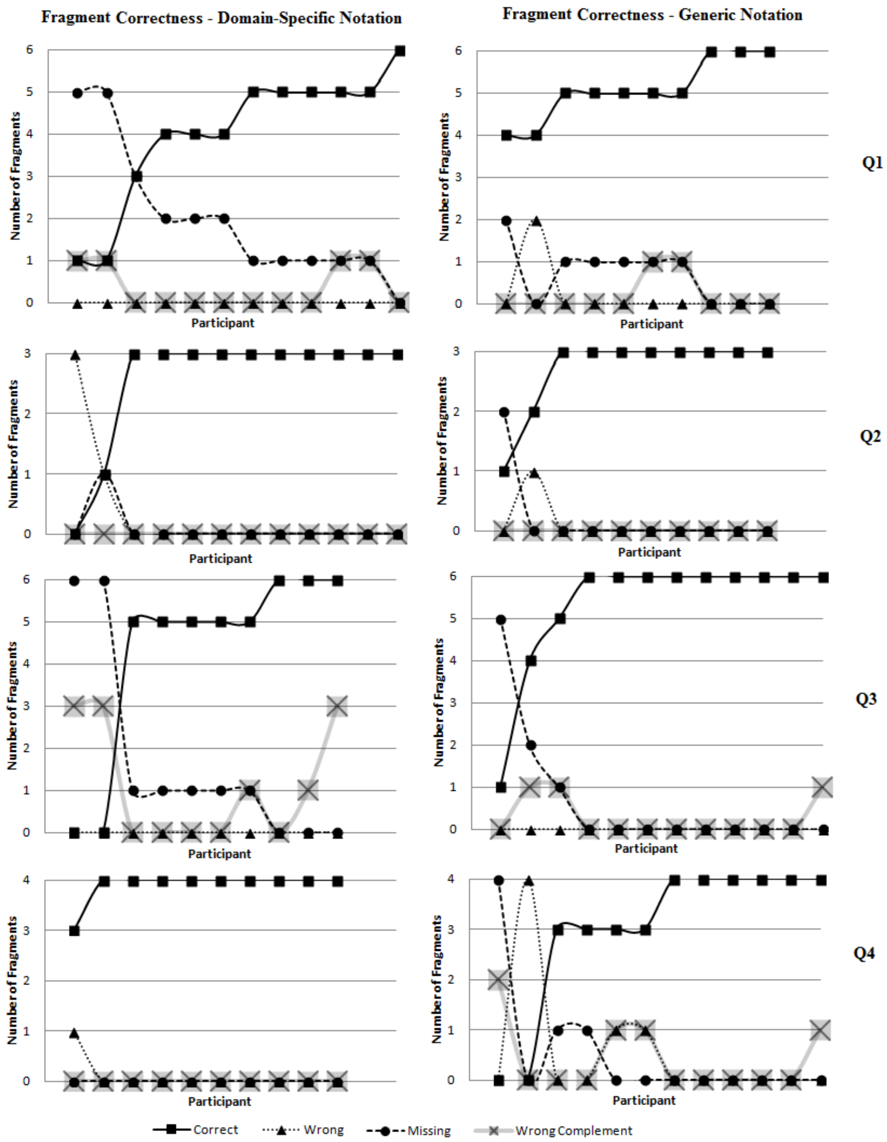


Fig. 6. Number of correct, wrong, missing and wrong complementary fragments per Question, for each Notation

4.2 Data Analysis and Discussion

According to our hypothesis, Group A should perform better on Q1 and Q2, while Group B should better perform on Q3 and Q4, as these are the times that each group works with the domain-specific notation. However, this was not always the case.

Looking at Table 2, we can say that, regarding response time, the expected results are confirmed for Q2, Q3 and Q4; however, the results contradicted our hypothesis for Q1. Moreover, the percentage differences between highest and lowest averages greatly varied. For Q1, highest and lowest averages had nearest values, while for Q4 they present the highest difference. The intention of generating such values is to observe how, on average, response times are different according to the notation. It is expected that the differences are significant, as occurred in Q2 and Q4, and even in Q3, favoring our hypothesis.

We have applied a statistical test, even having worked with a small sample. We applied the Wilcoxon-Mann-Whitney U Test [20], with significance level of 5%, for comparing response times. Considering groups A and B, U test indicated that the values are not significantly different, which is a good first indication that the groups are balanced. In Q1 and Q2, U test indicated that the values are not significantly different among groups, which was probably caused by the result of Q1. For Q3 and Q4, U test indicated that the values are significantly different among groups, what is a favorable result to our hypothesis.

However, an insulated analysis of response time is not enough. We need to check whether these results are corroborated by fragment correctness.

Looking at Table 3, which summarizes fragment correctness, we can notice that, again, we achieved results that are in favor and that contradict our hypothesis. Observing the number of correct fragments, we realize that specific notation worked better in Q4, but generic notation worked better in the other questions, with a small difference in Q2 (less than 1%). The most significant percentage of correctness occurred in Q4, and the lowest in Q1 (both from specific notation). Overall, the average correctness percentage is high (above 66%), demonstrating that participants had mostly success in the interpretation of models. On the number of wrong fragments, we had a slightly different result. In Q1 and Q4 we had fewer wrong fragments when using the specific notation, while in Q3 we had fewer wrong fragments when using the generic notation, in Q2 both had none error. Regarding missing fragments, the specific notation worked better for the objective questions (Q2 and Q4), while the generic notation worked better for the subjective questions (Q1 and Q3). Moreover, in the generic notation there is a relatively stable percentage (between 6% and 13%) of missing fragments. In the case of the specific notation, however, there is a stark difference when comparing objective and subjective questions. For objective questions, the percentages of missing fragments are very low (less than 3% for Q2, and 0% for Q4). On the other hand, for subjective questions, the percentages of missing fragments are very high (about 30% for Q1, and about 25% for Q3).

Finally, regarding the number of wrong complementary fragments, we can notice that, the specific notation worked better for the objective questions (Q4), while the generic notation worked better for the subjective questions (Q1 and Q3). We should also to highlight that, in Q2, there is not any wrong complementary fragment in both notations. Moreover, in the answers for Q3, when interpreting the model written in the specific notation, there is a high number of wrong complementary fragments.

Next, we complement our analysis presenting some detailed information for each question.

Question 1. The results obtained in this question clearly contradict our hypothesis. Q1 is a subjective question, requiring inspecting in details a model element. It requires a great attention by the participants, since it contains the greatest number of different fragments in the expected response. Response times were close, with a slight advantage for the generic notation. The missing parts stood out significantly in the domain specific notation. For instance, 10 participants did not indicate in the domain-specific notation that Lisa is a woman (83%), while only 6 participants in the generic notation did not indicate this fact (60%). Maybe it was considered obvious in the former notation, while in the case of the generic notation, perhaps the difficulty was lower, since this information is written in the model (allowing a textual perception). It is interesting to notice that, in fact, it was the number of missing fragments that caused the hypothesis contradiction, since the number of wrong fragments is favorable to the domain-specific notation. There were situations where the information was given only partially (counting as missing). For instance, there were an answer indicating that an employee supervises someone, but without indicating who is the supervised employee. Regarding the number of wrong complementary fragments, based on the answers, we suppose that some participants were not actually aware of the notation (commission treated as a kind of department or as a group).

Question 2. Q2 is an objective question. The domain-specific notation had a better performance regarding response time and the number of missing fragments. However, a better performance is achieved when using the generic notation with respect to the number of correct (a small difference) and wrong fragments. A possible explanation for this result came from the interpretation of a participant to this question: instead of answering who is the employee with the highest number of direct subordinates, she identified the employee with the greatest number of direct and indirect subordinates, giving rise to three wrong fragments. Once we worked with a small sample, this fact had a significant impact in the result. If we considered such participant an outlier, the result would have been reversed.

Question 3. Like Q1, this is a subjective question, and thus the results are similar: response times are close (with a small advantage for the domain-specific notation, as opposed to Q1), and there is a general advantage of the generic notation regarding correctness. Again, the number of missing fragments is able to change the outcome. In the specific notation, two participants indicated that there are 5 employees in the department, without indicating who are them (accounting for 10 missing fragments). In the generic notation, only one participant made this mistake. Moreover, in the specific notation, five participants did not indicate that the Marketing Department is part of the Administrative branch, while in the generic notation only one participant made this mistake. It is worthwhile to point out that the answers for this question presented the highest number of wrong complementary fragments in the experiment, highlighting the case of saying that Lisa is the leader of the Marketing Department (3 occurrences in generic notation and 5 occurrences in domain-specific notation).

Question 4. This was the question with the greatest proximity to our hypothesis. Specific notation presented better response time, higher percentage of correct fragments, and lower percentage of both wrong and missing fragments (in fact, just one error). This is a question that requires the participants realize which the members of a commission are, and then to which departments they belong. In the specific notation, this is easy to notice, since a person is within both the regions representing the commission and the department. On the other hand, in the generic notation, it is harder to follow the lines connecting the elements. Moreover, it is interesting to notice that the graph for this question when using the domain-specific notation is quite similar to the one for Q2, especially if we ignore the outlier in the latter.

In sum, the most prominent indication we noticed is that the participants using the domain-specific notation fared better on objective questions, while participants using the generic notation fared better in subjective questions. However, not all signs can be confirmed or justified. Nevertheless, we can say that the experiment fulfilled the objective of generating evidence in a qualitative way, being the starting point for subsequent experiments.

5 Final Considerations

In this paper we presented an empirical study aiming at collecting indications on the performance of participants in interpreting instance models, when using two different notations: a UML-based notation for object diagrams (said a generic notation), and a domain-specific visual language (DSVL), which was designed based on a well-founded ontology, following the approach discussed in [6]. In particular, we focus on the concrete syntax.

Some indications obtained from the results that need to be further explored: (i) Do the familiarity with the notation based on UML (generic one) may have assisted in the analysis of diagrams, closing, or even exceeding, the performance of the domain-specific notation, with which participants had the first contact? (ii) In the generic notation, all information about a given object are obtained in the same way: by navigating through links between objects, while with a DSVL, there are different ways of obtaining such information. How do this affect the results? (iii) Why in subjective questions is there a significant occurrence of missing fragments? Don't the participants perceive the information in the model, or at least do not feel the need to record the information, which can, for example, be considered "obvious"?

Future works will be directed to enhance indications identified here, trying to perform quantitative experiments, as well as deepening the knowledge regarding the unexpected data we collected. Some further experiments we foresee are: (i) to apply the experimental design discussed here in other domains; (ii) to apply the same experimental design but with participants with different profiles (novice modelers, model users). The first variation of the experiment aims at determining if the behavior is similar / different in a variety of domains. The second variation aims at identifying how the behavior is similar / different when we vary the participants' profile.

Acknowledgments. This research is funded by the Brazilian Research Funding Agencies FAPES (Process Number 52272362/11) and CNPq (Process Number #311578/2011-0) . We thank also to the participants of the experiment.

References

1. Mylopoulos, J.: Conceptual Modeling and Telos. In: Loucopoulos, P., Zicari, R. (eds.) Conceptual Modeling, Databases, and Case: An Integrated View of Information Sytems Development, pp. 49–68. Wiley (1992)
2. Avison, D.E., Fitzgerald, G.: Information Systems Development: Methodologies, Techniques and Tools. McGraw Hill, Oxford (2003)
3. Moody, D.L.: The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. IEEE TSE 35(6), 1–22 (2009)
4. Tolvanen, J.-P., Gray, J., Rossi, M.: Preface: A Special Issue on Domain-Specific Modeling with Visual Languages. Journal of Visual Languages and Computing (2004)
5. Gurr, C.A.: Effective Diagrammatic Communication: Syntactic, Semantic and Pragmatic Issues. Journal of Visual Languages and Computing 10(4), 317–342 (1999)
6. Guizzardi, G.: Ontology-Based Evaluation and Design of Visual Conceptual Modeling Languages. In: Reinhartz-Berger, I., Sturm, A., Clark, T., Cohen, S., Bettin, J. (eds.) Domain Engineering. Product Lines, Languages and Conceptual Models, p. 345. Springer, New York (2013)
7. Goodman, N.: Languages of Art: An Approach to a Theory of Symbols. Bobbs-Merril Co., Indianapolis (1968)
8. Guizzardi, G., Wagner, G., Guarino, N., van Sinderen, M.: An Ontologically Well-Founded Profile for UML Conceptual Models. In: Persson, A., Stirna, J. (eds.) CAiSE 2004. LNCS, vol. 3084, pp. 112–126. Springer, Heidelberg (2004)
9. Shimojima, A.: Operational Constraints in Diagrammatic Reasoning. In: Logical Reasoning with Diagrams, pp. 27–48 (1996)
10. Briguente, A.C., Falbo, R.A., Guizzardi, G.: Using a Foundational Ontology for Reengineering a Software Process Ontology. Journal of Information and Data Management (2012)
11. Gonçalves, B., Guizzardi, G., Filho, J.G.P.: Using an ECG Reference Ontology for Semantic Interoperability. Journal of Biomedical Informatics 43 (2010)
12. Guizzardi, G.: Ontological Foundations for Structural Conceptual Models. Telematica Instituut, Enschede, The Netherlands (2005)
13. Guizzardi, G., Wagner, G.: Using the Unified Foundational Ontology (UFO) as a Foundation for General Conceptual Modeling Languages. In: Theory and Application of Ontology. Springer, Berlin (2010)
14. Guizzardi, G.: Modal Aspects of Object types and Part-Whole Relations and the *de re/de dicto* distinction. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007. LNCS, vol. 4495, pp. 5–20. Springer, Heidelberg (2007)
15. Guizzardi, G.: The Problem of Transitivity of Part-Whole Relations in Conceptual Modeling Revisited. In: van Eck, P., Gordijn, J., Wieringa, R. (eds.) CAiSE 2009. LNCS, vol. 5565, pp. 94–109. Springer, Heidelberg (2009)
16. Ware, C.: Visual Thinking for Design. Morgan Kaufmann (2008)
17. Tversky, B., Hemenway, K.: Objects, Parts and Categories. Journal of Experimental Psychology 113, 169–193 (1984)
18. Moody, D., van Hillegersberg, J.: Evaluating the Visual Syntax of UML: An Analysis of the Cognitive Effectiveness of the UML Family of Diagrams. In: Gašević, D., Lämmel, R., Van Wyk, E. (eds.) SLE 2008. LNCS, vol. 5452, pp. 16–34. Springer, Heidelberg (2009)
19. Juristo, N., Moreno, A.M.: Basics of Software Engineering Experimentation. Springer Publishing Company, Incorporated (2001)
20. Wade, A., Koutoumanou, E.: Statistics Research & Methodology, `https://epilab.ich.ucl.ac.uk/coursematerial/statistics/index.html`