

Challenges of Identifying Communities with Shared Semantics in Enterprise Modeling

Dirk van der Linden^{1,2,4} and Stijn Hoppenbrouwers^{3,2,4}

¹ Public Research Centre Henri Tudor, Luxembourg, Luxembourg
`dirk.vanderlinden@tudor.lu`

² Radboud University Nijmegen, Nijmegen, the Netherlands

³ HAN University of Applied Sciences, Arnhem, the Netherlands
`stijn.hoppenbrouwers@han.nl`

⁴ EE-Team, Luxembourg, Luxembourg*

Abstract. In this paper we discuss the use and challenges of identifying communities with shared semantics in Enterprise Modeling. People tend to understand modeling meta-concepts (i.e., a modeling language’s constructs or types) in a certain way and can be grouped by this understanding. Having an insight into the typical communities and their composition (e.g., what kind of people constitute a semantic community) would make it easier to predict how a conceptual modeler with a certain background will generally understand the meta-concepts he uses, which is useful for e.g., validating model semantics and improving the efficiency of the modeling process itself. We demonstrate the use of psychometric data from two studies involving experienced (enterprise) modeling practitioners and computing science students to find such communities, discuss the challenge that arises in finding common real-world factors shared between their members to identify them by and conclude that the common (often implicit) grouping properties such as similar background, focus and modeling language are not supported by empirical data.

Key words: enterprise modeling, conceptual understanding, personal semantics, community identification, semantics clustering

1 Introduction

The modeling of an enterprise typically comprises the modeling of many aspects (e.g., processes, resources, rules), which themselves are typically represented in a specialized modeling language or method (e.g., BPMN [1], e3Value [2], RBAC [3]). Most of these languages share similar meta-concepts (e.g., PROCESSES, RESOURCES, RESTRICTIONS⁵). However, from language to language (and modeler to modeler) the way in which these meta-concepts are typically used

* The Enterprise Engineering Team (EE-Team) is a collaboration between Public Research Centre Henri Tudor, Radboud University and HAN University of Applied Sciences (www.ee-team.eu)

⁵ To distinguish concepts from words used for them we print concepts in SMALL CAPS.

(i.e., their intended semantics) can differ. For example, one modeler might typically intend RESTRICTIONS to be deontic in nature (i.e., open guidelines that ought to be the case), while a different modeler might typically consider them as alethic conditions (i.e., rules that are strict logical necessities). They could also differ in whether they typically interpret RESULTS as being material or immaterial ‘things’. If one is to integrate or link such models (i.e., the integrative modeling step in enterprise modeling [4, 5, 6, 7]) and ensure the consistency and completeness of the involved semantics, it is necessary to be aware of the exact way in which such a meta-concept was used by the modeler. If this is not explicitly taken into account, problems could arise from, e.g., treating superficially similar concepts as being the same or eroding the nuanced view from specific models when they are combined and made (internally) consistent.

To deal more effectively with such semantic issues it is necessary to have some insight into the “mental models” of the modeler. It is important to gain such insight because people generally do not think in the semantics of a given modeling language, but in the semantics of their own natural language [8]. Furthermore, some modeling languages do not have an official, agreed-upon specification of their semantics [9] and if they do, there is no guarantee that their semantics are complete or consistent [10, 11, 12], let alone that users might deliberately or unconsciously ignore the official semantics and invent their own [13]. Understanding the intended semantics of a given model thus can not come only from knowledge of the language and its semantics, but requires us to spend time understanding the modeler who created the model.

However, one cannot realistically be expected to look into each individual modeler’s semantic idiosyncrasies. Instead, a generalized view on how people with a certain background typically understand the common meta-concepts could be used to infer, to some degree of certainty, the outline of their conceptual understanding. Such (stereo)types of modelers could be found by identifying communities of modelers that share similar semantic tendencies for given concepts and analyzing whether they have any shared properties that allow us to treat them as one. As language itself is inherently the language of community [14] (regardless of whether that community is bound by geography, biology, shared practices and techniques [15] or simply speech and natural language [16, 17]), it is safe to assume that there are communities which share a typical way of using (and understanding) modeling language concepts. This is not to say that such communities would be completely homogeneous in their semantics, but merely that they contain enough overlap to be able to be treated as belonging together during a process which integrates models originating from their members without expecting strong inconsistencies to arise in the final product.

Finding such communities based on, for example, empirical data is not a difficult matter in itself. However, going from simply finding communities to understanding them and generalizing them (i.e., to be able to predict on basis of empirical data or prior experience that communities of people which share certain properties will typically have certain semantics) is the difficult step. To do so it is necessary to find *identifiers* – properties that are shared between the members

of a community. These *identifiers* (e.g., dominant modeling language, focus on specific aspects) are needed to be able to postulate that a given modeler, with a given degree of certainty, belongs to some community and thus likely shares this community's typical understanding of a concept.

In workshop sessions held with companies and practitioners from the Agile Service Development (ASD)⁶ project who are involved in different kinds of (collaborative) domain modeling (e.g., enterprise modeling, knowledge engineering, systems analysis) we have found that there are a number of common identifiers modelers are typically (and often implicitly) grouped by. That is, on basis of these properties they are often assigned to collaborate on some joint domain modeling task. These properties are for example a similar background, education, focus on what aspects to model (e.g., processes, goals) in what sector they do so (e.g., government, health care, telecommunications), and used modeling languages. It seems thus that in practice, it is assumed that those who share a background or use similar modeling languages and methods will model alike.

While the wider context of our work is to build towards a theory of how people understand typical modeling meta-concepts (which can aid enterprise modelers with creating integrated models) this paper will focus first on testing the above assumption. To do so we hypothesize that these commonly used properties (e.g., sector, focus, used modeling language) should be reflected in communities that share a similar semantic understanding of common modeling meta-concepts. To test this we will investigate the personal semantics for practitioners and students alike, group them by shared semantics and investigate whether they share these, or indeed, any amount properties. If this is found to be so, it could mean that it is possible to predict, to a certain degree, what (the range of) understanding is that a modeler has for a given concept.

In this stage of our empirical work we have enough data from two of our studies into the conceptual understanding of the common meta-concepts amongst practitioners and students (cf. [18] for some initial results) to have found several communities that share a similar understanding of conceptual modeling meta-concepts. However, we have begun to realize the difficulties inherent in properly *identifying* them. The rest of this paper is structured as follows. In Section 2 we discuss the used data and how we acquired it. In Section 3 we demonstrate how this kind of data can be analyzed to find communities, discuss the difficulties in identifying common properties amongst their members and reflect on the hypothesis. Finally, in Section 4 we conclude and discuss our future work.

⁶ The ASD project (www.novay.nl/okb/projects/agile-service-development/7628) was a collaborative research initiative focused on methods, techniques and tools for the agile development of business services. The ASD project consortium consisted of Be Informed, BiZZdesign, Everest, IBM, O&i, PGGM, RuleManagement Group, Voogd & Voogd, CRP Henri Tudor, Radboud University Nijmegen, University Twente, Utrecht University & Utrecht University of Applied Science, TNO and Novay.

2 Methods and Used Data Samples

The data used in this paper originates from two studies using semantic differentials into the personal semantics participants have for a number of meta-concepts common to modeling languages and methods used in Enterprise Modeling. The Semantic Differential [19] is a psychometric method that can be used to investigate what connotative meanings apply to an investigated concept, e.g., whether an investigated concept is typically considered good or bad, intuitive or difficult. It is widely used in information systems research and there are IS-specific guidelines in order to ensure quality of its results [20]. We use semantic differentials to investigate the attitude participants have towards ACTORS, EVENTS, GOALS, PROCESSES, RESOURCES, RESTRICTIONS and RESULTS and to what degree they can be considered *natural*, *human*, *composed*, *necessary*, *material*, *intentional* and *vague* things. These concepts and dimensions originate from our earlier work on categorization of modeling language constructs [21]. The resulting data is in the form of a matrix with numeric scores for each concept-dimension combination (e.g., whether an ACTOR is a *natural thing*, whether a RESULT is a *vague thing*). Each concept-dimension combination has a score ranging from 2 to -2, denoting respectively agreement and disagreement that the dimension ‘fits’ with their understanding. A more detailed overview of the way we apply this method is given in [22].

The practitioner data sample ($n=12$) results from a study which was carried out in two internationally operating companies focused on supporting clients with (re)design of organizations and enterprises. The investigated practitioners all had several years of experience in applying conceptual modeling techniques. We inquired into the modeling languages and methods they use, what sector(s) they operate in, what they model, and what kind of people they mostly interact with. The student data sample ($n=19$) results from an ongoing longitudinal study into the (evolution of) understanding computing and information systems science students have of modeling concepts. This study started when the students began their studies and had little to no experience. We inquired into their educational (and where applicable professional) background, knowledge of modeling or programming languages and methods, interests and career plans in order to see whether these could be used as identifying factors for a community.

To find communities of people that share semantics (i.e., score similarly for a given concept) we analyzed the results using Repeated Bisection clustering and Principal Component Analysis (PCA). The PCA results and their visualization (see Figs. 2 and 1) demonstrate (roughly) the degree to which people share a (semantically) similar understanding of the investigated concepts (for the given dimensions) and can thus be grouped together.

3 General Results & Discussion

Most importantly, the results support the idea that people can be clustered based on their personal semantics. The PCA data proved to be a more useful

resource for investigating the clusters and general semantic distance than the (automated) clustering itself, as we found it was hard to a priori estimate parameters like optimal cluster size and similarity cutoffs. As shown in Figs. 1 and 2 there are easily detectable clusters (i.e., communities) for most of the investigated concepts, albeit varying in their internal size and variance. The closer two participants are on both axes, the more similar (the quantification of) their semantics are.

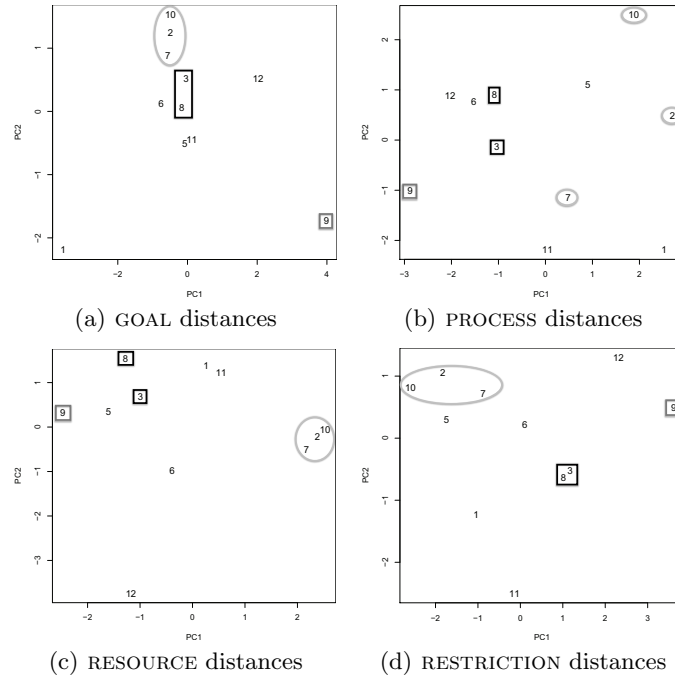


Fig. 1. Principal components found in the data of concept-specific understanding for practitioners. The visualizations represent (roughly) the distance between understandings individual participants have. The further away two participants are on both axes (i.e., horizontal and vertically different coordinates), the more different their conceptual understanding has been measured to be. Shown are the distances between participants for their understanding of GOALS, PROCESSES, RESOURCES and RESTRICTIONS, with some discussed participants highlighted. Colored boxes and circles are used to highlight some interesting results that will be discussed in more detail.

While there are both clusters of people that share a semantic understanding for students and practitioners alike, they do differ to what degree larger clusters can be found. Internal variance is generally greater for students (i.e., the semantics are more ‘spread out’). This may be explained by the greater amount of neutral attitudes practitioners display towards most of the dimension (i.e., lack any strongly polarized attitudes) causing a lower spread of measurable seman-

tics. Such neutral attitudes might be a reflection of the necessity to be able to effectively interact with stakeholders who hold different viewpoints. Nonetheless, they are still easily divided into communities based on their semantic differences.

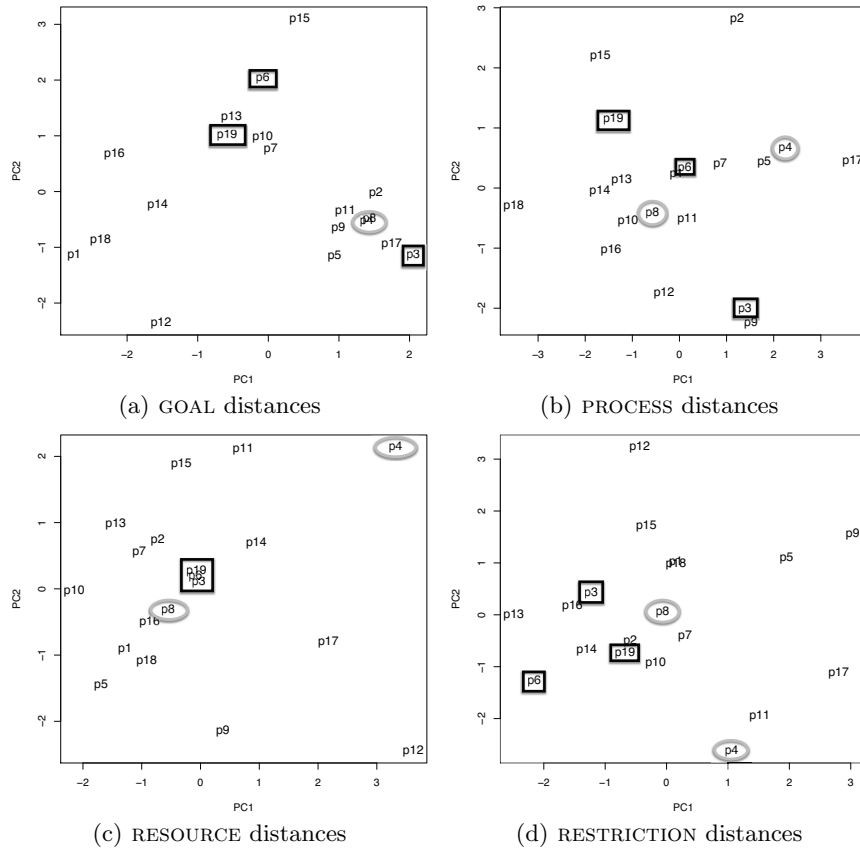


Fig. 2. Principal components found in the data of concept-specific understandings for students. The visualizations represent (roughly) the distance between understandings individual participants have. The further away two participants are on both axes (i.e., horizontal and vertically different coordinates), the more different their conceptual understanding has been measured to be. Shown are the distances between for participants for their understanding of GOALS, PROCESSES, RESOURCES and RESTRICTIONS, with some discussed participants highlighted.

To demonstrate, we will discuss some of the clusters we found for the understanding practitioners and students have of GOALS, PROCESSES, RESOURCES and RESTRICTIONS. The immediately obvious difference between the practitioners and students is that, where there are clusters to be found amongst the practitioners, they differ mostly on one axis (i.e., component), whereas the students often differ wildly on both axes. Of particular interest to testing our hypothesis

are participants 3 & 8, and 2, 7 & 10 from the practitioner data sample. The first community clusters together very closely for their understanding of RESTRICTIONS (and GOALS, albeit to a lesser degree) while they differ only slightly for most other concepts. This means one would expect them to share some real-world properties. Perhaps they are people specialized in goal modeling, or share a typical way of modeling RESTRICTIONS in a formal sense. The second community (participants 2, 7 & 10) cluster together very closely for RESOURCES, fairly close for GOALS and RESTRICTIONS, while being strongly different when it comes to their understanding of PROCESSES. One could expect this to infer that they have some shared focus on RESOURCES, either through a language they use (e.g., value-exchange or deployment languages) which are often strongly connected to GOALS (as either requiring them, or resulting in their creation). On the opposite, one would not necessarily expect there to be much overlap between the participants in regards to PROCESSES, as they are grouped with a wide spread.

For the students, there are several potentially interesting communities to look at. Participants 4 & 8 differ strongly for several concepts (e.g., their strong differentiation on two components for RESOURCES, and for PROCESSES and RESTRICTIONS), but they have an almost exactly similar understanding of GOALS. One would expect that some kind of property shared between them might be used to identify other participants that cluster together for GOALS, but not necessarily share other understandings. Participants 3, 6 & 19 also cluster together closely for one concept – RESOURCES – but differ on their understanding of the other investigated concepts. As such, if (some) experience in the form of having used specific programming and modeling languages is correlated to their conceptual understanding, one would expect to find some reflection of that in the clusterings of these students.

However, when we add the information we have about the participants (see Tables 1 and 2) to these clusters, we run into some problems. It is often the case that communities do not share (many) pertinent properties, or when they do, there are other communities with the same properties that are far removed from them in terms of their conceptual understanding. Take for instance participants 2, 7 & 10 (highlighted with a gray oval) from the practitioner data sample. While they share some properties, (e.g., operating in the same sector, having some amount of focus on PROCESSES, and interacting with domain experts), when we look at other communities it is not as simple to use this combination of properties to uniquely identify them. For instance, participants 3 & 8 (highlighted with a black rectangle) cluster together closely in their own right, but do share some overlapping properties (both operate in the government sector). Thus, merely looking at the sector a modeler operates in cannot be enough to identify them. Looking at the combination of sector and focus is not enough as well, as under these conditions participant 8 and 10 should be grouped together because they both have a focus on rules. When we finally look at the combination of sector, focus and interaction we have a bit more chance of uniquely identifying communities, although there are still counter-examples. Participant 9 (highlighted with a gray rectangle), for example, shares all the properties with

participants 2, 7 & 10, but is conceptually far removed from all others. In general the dataset shows this trend, providing both examples and counterexamples for most of these property combinations, making it generally very difficult, if not flat-out impossible to identify communities.

Table 1. Comparison of some practitioners based on investigated properties. The proprietary language is an in-house language used by one of the involved companies.

No.	Used languages	Sector	Focus	Interacts with
3	Proprietary	Financial, Government	Knowledge processes, data	rules, Analysts, modelers
8	UML, OWL, RDF, Mindmap, Proprietary	Government, Healthcare	Rules	Business professionals, policymakers, lawyers
2	Proprietary	Government	Knowledge systems, processes	Managers, domain experts
7	Proprietary, Java	UML, Government, spatial planning	Business process structure	Domain experts, IT specialists
10	Proprietary, xml, xslt	Government, finance	Processes, object for systems	rules, Domain experts, java developers
9	Proprietary	Government, Financial	Rules, policy, processes	legislation, Domain experts

We face the same challenge in the student data sample, although even more pronounced on an almost individual level. There are participants that share the same properties while having wildly varying conceptual understandings. There seems to be some differentiation on whether participants have prior experience, but even then this sole property does not have enough discriminatory power. Take for example participants 4&8 (highlighted with a black rectangle) and participants 3,6&19 (highlighted with a gray oval). Both these communities cluster closely together for a specific concept, but then differ on other concepts. One could expect this has to do with a small amount of properties differing between them, which is the case, as there is consistently a participant with some prior experience in programming and scripting languages amongst them. However, if this property really is the differentiating factor, one would expect that on the other concepts the participants with prior experience (4&6) would be further removed from other participants than the ones without experience are, which is simply not the case. It thus seems rather difficult to link these properties to the communities and their structure.

This challenge could be explained by a number of things. First and foremost would be a simple lack in the amount of properties (or their granularity, as might be the case in the student data sample) to identify communities by, while it is

Table 2. Comparison of some students based on investigated properties. Profiles are standardized packages of coursework students took during secondary education, nature being natural sciences, technology a focus on physics and health a focus on biology.

No. Study	Profile	Prior experience
4 Computing Science	Nature & Technology & Health	Some programming and scripting experience
8 Computing Science	Nature & Technology	None
3 Information Systems	Nature & Technology	None
6 Computing Science	Nature & Technology	Programming experience
19 Information Systems	Nature & Health	None

also possible that the investigated concepts were not at the right abstraction level (i.e., either too specific or too vague), or that the investigated concepts were simply not the concepts people use to model.

The simplest explanation is that the properties we attempt to identify communities by are not the right (i.e., properly discriminating) ones. It is possible (especially for the student data sample) that some of the properties are not necessarily not right, but that they are not discriminative enough. For example, knowing what modeling languages someone uses could be described in more detail because a language could have multiple versions that are in-use, and it is possible (indeed quite likely) that a language as-used is not the same as the ‘official’ language. However, this line of reasoning is problematic for two reasons. The first being that these are properties that are used by practitioners to (naively) group modelers together, the second that there is no clear-cut way to identify reasonable other properties that are correlated to the modeling practice. If these properties are not useful, we would have to reject the hypothesis on grounds of them being a ‘bad fit’ for grouping people. Other properties that could be thought of could include reflections of the cultural background of modelers, however, these are less likely to be of influence in our specific case as the Enterprise Modelers we investigate are all set in a Western European context and there is little cultural diversity (or their granularity, as might be the case in the student data sample) in this sense.

Another explanation could be that the meta-concepts we chose are not at the right abstraction level (i.e., concept width), meaning that they are either too vague or specific. For example, some modelers could typically think on near-instantiation level while others think more vague. If concepts are very specific one would actually expect to find differences much faster (as the distance between people’s conceptual understanding can be expected to be larger), which thus makes it *easier* to find communities. If they are (too) vague though, people would not differ much because there are not enough properties to differ on in the first place. However, the way we set up our observations rules out the vagueness possibility, as participants were given a semantic priming task before the

semantic differential task of each concept. What we investigated was thus their most typical specific understanding of a concept. For this reason it is unlikely the abstraction level of the concepts was the cause of the challenge of identifying the communities.

Finally, the most obvious explanation could be a flaw in our preliminary work, namely that we did not select the right concepts, irrespective of their abstraction level. Considering the concepts were derived from an analysis of conceptual modeling languages and methods used for many aspects of enterprises, and that there simply does not seem a way to do without most of them, we find it very unlikely this is the case. The unlikely option that what we investigated was not actually the modeling concept, but something completely else (i.e., someone considering their favorite Hollywood actors over a conceptual modeling interpretation of ACTOR) can also be ruled out as the priming task in our observation rules out this possibility. It thus seems far more prudent that these potential issues did not contribute to the challenge we face, and we should move towards accepting that identifying communities of modelers based on the investigated properties might not be a feasible thing to do.

While we had hoped that these observations would have yielded a positive result to the hypothesis, the lack of support we have shown means that a theory of predicting how modelers understand the key concepts they use, and thus what the additional ‘implicit’ semantics of a model could be (as alluded to earlier) is likely not feasible. Nonetheless, the observations do help to systematically clarify that these different personal understandings exist, can be measured, and might be correlated to communication and modeling breakdown due to unawareness of linguistic prejudice. Eventually, in terms of Gregor’s [23] types of theories in information systems this information can be used by enterprise modelers and researchers alike to build design theories supporting model integration in enterprise modeling by pointing out potentially sensitive aspects of models’ semantics.

If we wanted to simply discount the possibility of these properties being good ways to identify communities that share a semantic understanding of some concepts with, we would be done. But there is more of an issue here, as these properties *are* being used to identify communities and group people together in practice. Thus, given these findings we have to reject the hypothesis as stated in our introduction, while as of yet not being able to replace it with anything but a fair warning and call for more understanding – do not just assume (conceptual) modelers will model alike just because they have been using the same languages, come from the same background or work in the same area.

To summarize, we have shown that the often implicit assumption that people have strongly comparable semantics for the common modeling meta-concepts if they share an expertise in certain sectors, modeling focus and used languages cannot be backed up by our empirical investigation. While not an exhaustive disproof of the hypothesis by any means, it casts enough doubt on it that it would be a considerate practice for Enterprise Modelers to be more careful and double-check their assumptions when modeling together with, or using models from, others practitioners.

4 Conclusion and Future Work

We have shown a way to discover communities that share semantics of conceptual modeling meta-concepts through analysis of psychometric data and discussed the difficulties in identifying them through shared properties between their members. On basis of this we have rejected the hypothesis that modelers with certain shared properties (such as used languages, background, focus, etc.) can be easily grouped together and expected to share a similar understanding of the common conceptual modeling meta-concepts.

Our future work involves looking at the used properties in more detail (i.e., what exactly a used language constitutes) and a more detailed comparison of the results of practitioners and students in terms of response polarity and community distribution. Furthermore we will investigate whether there is a correlation between the specific words that a community typically uses to refer to its concepts.

Acknowledgments. This work has been partially sponsored by the *Fonds National de la Recherche Luxembourg* (www.fnrl.lu), via the PEARL programme.

References

1. Object Management Group: Business process model and notation (bpmn) ftf beta 1 for version 2.0. Internet (2010)
2. Gordijn, J., Yu, E., van der Raadt, B.: e-service design using i* and e3value modeling. *IEEE Software* **23** (2006) 26–33
3. Ferrariolo, D., Cugini, J., Kuhn, R.: Role-based access control (rbac): Features and motivations. In: Proc. of the 11th Annual Computer Security Applications Conference. (1995)
4. Lankhorst, M.M.: Enterprise architecture modelling—the issue of integration. *Advanced Engineering Informatics* **18**(4) (2004) 205 – 216
5. Kuehn, H., Bayer, F., Junginger, S., Karagiannis, D.: Enterprise Model Integration. In: E-Commerce and Web Technologies. Volume 2738 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2003) 379–392
6. Vernadat, F.B.: Enterprise modeling and integration (EMI): Current status and research perspectives. *Annual Reviews in Control* **26**(1) (2002) 15 – 25
7. Opdahl, A.L., Berio, G.: Interoperable language and model management using the UEMML approach. In: Proceedings of the 2006 International Workshop on Global Integrated Model Management, New York, NY, USA, ACM (2006) 35–42
8. Sowa, J.: The Role of Logic and Ontology in Language and Reasoning. In: Theory and Applications of Ontology: Philosophical Perspectives. Springer Netherlands (2010) 231–263
9. Ayala, C.P., Cares, C., Carvallo, J.P., Grau, G., Haya, M., Salazar, G., Franch, X., Mayol, E., Quer, C.: A comparative analysis of i*-based agent-oriented modeling languages. In: SEKE’05, Taipei, Taiwan. (2005) 43–50
10. Breu, R., Hinkel, U., Hofmann, C., Klein, C., Paech, B., Rumpe, B., Thurner, V.: Towards a formalization of the Unified Modeling Language. In M. Aksit et al., ed.: ECOOP’97 Object-Oriented Programming. Volume 1241 of LNCS. Springer Berlin / Heidelberg (1997) 344–366 10.1007/BFb0053386.

11. Nuffel, D., Mulder, H., Kervel, S.: Enhancing the formal foundations of bpmn by enterprise ontology. *Advances in Enterprise Engineering III* (2009) 115–129
12. Wilke, C., Demuth, B.: UML is still inconsistent! How to improve OCL Constraints in the UML 2.3 Superstructure. *Electronic Communications of the EASST* **44**(0) (2011)
13. Henderson-Sellers, B.: UML - the Good, the Bad or the Ugly? Perspectives from a panel of experts. *Software and System Modeling* **4**(1) (2005) 4–13
14. Perelman, C., Olbrechts-Tyteca, L.: *The New Rhetoric: A Treatise on Argumentation*. University of Notre Dame Press (June 1969)
15. Wenger, E., Snyder, W.: Communities of practice: The organizational frontier. *Harvard business review* **78**(1) (2000) 139–146
16. Gumperz, J.: The speech community. In: *Linguistic anthropology: A reader*. Wiley-Blackwell (2001) 66–74
17. Hoppenbrouwers, S.J.B.A.: Freezing language : conceptualisation processes across ICT-supported organisations. PhD thesis, Radboud University Nijmegen (2003)
18. van der Linden, D., Gaaloul, K., Molnar, W.: Initial results from a study on personal semantics of conceptual modeling languages. *Natural Language Processing and Information Systems* (2012) 360–365
19. Osgood, C.E., Suci, G.J., Tannenbaum, P.: *The Measurement of Meaning*. University of Illinois Press, Urbana (1957)
20. Verhagen, T., Meents, S.: A framework for developing semantic differentials in is research: Assessing the meaning of electronic marketplace quality (emq). *Serie Research Memoranda 0016*, VU University Amsterdam, Faculty of Economics, Business Administration and Econometrics (2007)
21. van der Linden, D.J.T., Hoppenbrouwers, S.J.B.A., Lartseva, A., Proper, H.A.: Towards an investigation of the conceptual landscape of enterprise architecture. In T. Halpin et al., ed.: *Enterprise, Business-Process and Information Systems Modeling*. Volume 81 of LNCS. Springer Berlin Heidelberg (2011) 526–535
22. van der Linden, D., Hoppenbrouwers, S., Lartseva, A., Molnar, W.: Beyond terminologies: Using psychometrics to validate shared ontologies. Accepted for publication in *Applied Ontology* (February 2012)
23. Gregor, S.: The nature of theory in information systems. *Mis Quarterly* **30**(3) (2006) 611–642