On the Identification of Modeler Communities

Dirk van der Linden, Public Research Centre Henri Tudor, Luxembourg, Luxembourg & Radboud University Nijmegen, Nijmegen, Netherlands & EE-Team, Luxembourg

Stijn J.B.A. Hoppenbrouwers, HAN University of Applied Sciences, Arnhem, Netherlands & Radboud University Nijmegen, Nijmegen, Netherlands & EE-Team, Nijmegen, Netherlands

Henderik A. Proper, Public Research Centre Henri Tudor, Luxembourg, Luxembourg & Radboud University Nijmegen, Nijmegen, Netherlands & EE-Team, Luxembourg

ABSTRACT

The authors discuss the use and challenges of identifying communities with shared semantics in Enterprise Modeling (EM). 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 conceptual understanding. Having an insight into the typical communities and their composition (e.g., what kind of people constitute such a semantic community) can make it easier to predict how a conceptual modeler with a certain background will generally understand the meta-concepts s/he uses, which is useful for e.g., validating model semantics and improving the efficiency of the modeling process itself. The authors have observed that in practice decisions to group people based on certain shared properties are often made, but are rarely backed up by empirical data demonstrating their supposed efficacy. The authors demonstrate the use of psychometric data from two studies involving experienced (enterprise) modeling practitioners and computing science students to find such communities. The authors also discuss the challenge that arises in finding common real-world factors shared between their members to identify them by and conclude that there is no empirical support for commonly used (and often implicit) grouping properties such as similar background, focus and modeling language.

Keywords: Conceptual Understanding, Enterprise Modeling, Meta-Concepts, Modeling Concepts, Modeling Languages

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 (Object Management Group, 2010), e3Value (Gordijn et al., 2006), RBAC (Ferrariolo et al., 1995)). Most of these languages share similar meta-concepts (e.g., PROCESSES, RESOURCES, RESTRICTIONS). However, from language to language (and modeler to
modeler) the way in which meta-concepts are typically used (i.e., their intended semantics) can differ. For example, one modeler might typically intend restrictions to be deontic in nature (i.e., restrictions that ought to be the case, but can be violated), while a different modeler might typically consider them as alethic conditions (i.e., rules that are strict logical necessities and cannot be violated). The modelers could also differ in whether they typically interpret results as being material or immaterial ‘things’. Even for scenarios as simple as the delivery of a pizza these differences become apparent, as a pizza delivery can include alethic restrictions in order to observe temporal dependencies (“A pizza cannot be delivered before it is made.”), deontic restrictions (“A pizza should be delivered within 30 minutes of its order.”), and the result of the delivery can be a material thing (a certain amount of notes and coins of the local currency) or an immaterial one (a confirmation of payment on a debit card machine). If one is to integrate or link models (i.e., the integrative modeling step in enterprise modeling (cf. Lankhorst, 2004; Kuehn et al., 2003; Vernadat, 2002; Opdahl and Berio, 2006; Delen et al., 2005) 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.

This challenge follows from the collaborative nature of enterprise modeling (cf. Ssebaggawo et al., 2009; Rospocher et al., 2008; Frederiks and van der Weide, 2006; Hoppenbrouwers et al., 2005; Hoppenbrouwers et al., 2006), as it involves different people specialized in different aspects of the enterprise. These aspects have to be elaborated on to deal with the complexity of (re)designing modern day enterprises (Barjis et al., 2009). Collaborative modeling in general (cf. Rouwette et al., 2008; Hoppenbrouwers et al., 2009; Rittgen, 2009) deals with challenges like these that arise because of the different people involved, such as optimizing the actual modeling process (Bidarra et al., 2001), ensuring its effectiveness (Dean et al., 2000) and dealing with conflicts and problems that arise when integrating models made by different people with different viewpoints (Renger et al., 2008).

The particular challenge we are concerned with in the enterprise modeling process is mismatched understandings between different modelers and stakeholders (Kaidalova et al., 2012). Note that mismatched understanding does not only refer to misunderstandings that the involved parties might be aware of. It explicitly also refers to the (more damaging) misunderstanding that the parties involved might not be aware of.

People might disagree on what words to use, what they should mean, or use the same words without realizing they talk about different things. When these apparent or hidden disagreements extend to the words used by a modeling language (i.e., the meta-concepts), the produced models themselves might no longer reflect correctly or fully the conceptualization of the individuals involved. As models should be there to support the building of knowledge and its exchange (Stahl, 2000), any threat to the validity and integrity of the models is a threat to the knowledge exchange itself. An often-used strategy to deal with this is a priori agreeing on or working towards a set of standardized terminology and semantics. However, it is neither safe nor effective to simply assume that such expressed agreements, or even the models themselves, express correctly and completely the way a modeler conceptualizes them (Guarino et al., 1994).

To deal more effectively with the issue of semantics it is necessary to have an insight into the ‘mental models’ of the people involved (Uschold, 2011; Almeida, 2009). It is important to gain such insight on a personal level because “semantic memory for concepts is based on a subject’s memories of past experiences with instances of those concepts” (Geeraerts, 2010) and because people generally do not think in the semantics of a given modeling language, but
An Extension of Business Process Model and Notation for Security Risk Management

Software Development Using Service Syndication Based on API Handshake Approach between Cloud-Based and SOA-Based Reusable Services
[www.igi-global.com/chapter/software-development-using-service-syndication/65170?camid=4v1a](www.igi-global.com/chapter/software-development-using-service-syndication/65170?camid=4v1a)