Information Systems 37 (2012) 213–226

Contents lists available at SciVerse ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/infosys

"Modeling with tools is easier, believe me"—The effects of tool functionality on modeling grammar usage beliefs

Jan Recker*

Queensland University of Technology, 126 Margaret Street, QLD 4000, Brisbane, Australia

ARTICLE INFO

Article history:
Received 2 June 2010
Received in revised form
21 October 2011
Accepted 24 October 2011
Recommended by: P. Loucopoulos
Available online 6 November 2011

Keywords:
Conceptual modeling
Perception measurement
Usage beliefs
Tool functionality

ABSTRACT

Increasingly, studies are reported that examine how conceptual modeling is conducted in practice. Yet, typically the studies to date have examined in isolation how modeling grammars can be, or are, used to develop models of information systems or organizational processes, without considering that such modeling is typically done by means of a modeling tool that extends the modeling functionality offered by a grammar through complementary features. This paper extends the literature by examining how the use of seven different features of modeling tools affects usage beliefs users develop when using modeling grammars for process modeling. We show that five distinct tool features positively affect usefulness, ease of use and satisfaction beliefs of users. We offer a number of interpretations about the findings. We also describe how the results inform decisions of relevance to developers of modeling tools as well as managers in charge for making modeling-related investment decisions.

1. Introduction

Information systems (IS) analysts and designers need to have an understanding about the domain in which the system is meant to operate, and the functions it has to perform [43]. To address this task, they often create conceptual models of the relevant business domains the information system is intended to support. These models are created using semi-formal, diagrammatic conceptual modeling grammars that provide graphical constructs and rules how to combine these constructs [68].

Conceptual modeling is an active research area in Information Systems [11]. Related research has examined, for instance, how conceptual modeling grammars are capable of creating models that provide a faithful representation of some real world domain [62], how a specific conceptual model provides a faithful representation of a real world domain [51], or, more generally, how the quality of conceptual models can be managed [47].

Recently, research has started to examine how conceptual modeling grammars are used in practice. It was shown, for instance, that direct utility beliefs (such as ease of use, usefulness and satisfaction) are key determinants of users' decisions to continue to use a modeling grammar [54]. It was also shown that perceived usefulness and perceived ease of use of a modeling grammar are dependent on their ontological properties; viz., their levels of construct deficit, redundancy, overload and excess [61]. Other studies have examined, for example, different usage patterns of conceptual modeling grammars [20,72].

Our interest in this paper is to extend the current body of knowledge about grammar usage beliefs in conceptual modeling. Research to date has examined how models are created using grammars or methods (e.g., [67]), how the models are understood [51] or used in a variety of settings [1,50]. Yet, the usage of these modeling artifacts has typically been studied *in isolation*, i.e., decoupled from the modeling environment in which modeling is conducted.

Specifically, most studies that examined the use of modeling grammars have not explicitly considered that these grammars are typically implemented, and used, within a modeling tool. These tools have become very

* Tel.: +61 731389479.
E-mail address: j.recker@qut.edu.au

0306-4379/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.is.2011.10.007

sophisticated and provide extended functionality to support the way grammars can be deployed. For instance, some tools provide model repositories in which models can be stored and cross-linked on different levels of conceptual abstraction [41]. Also, most tools offer a variety of grammars to use for conceptual modeling, which, in turn, enables users to overcome any type of deficiency they might encountered in any given grammar [31]. Finally, more recently, tools have emerged that provide collaboration support for modeling, for instance, through advanced visualization features [7].

Our primary conjecture is, therefore, that modeling tools impact the way that modeling grammars are used by analysts. *How* exactly modeling tools affect usage of modeling grammars, however, is still unclear. Moreover, it remains unclear which modeling tool functionality specifically affects the way that grammars are used.

In this paper, therefore, we report on research we undertook to examine the effects of seven types of tool functionality on three key grammar usage beliefs. We draw on data collected as part of a large field study [54–56] of users of the Business Process Modeling (BPMN) grammar [6], the current industry standard in process-aware modeling of information systems; and examine the data collected with a specific focus on the reported tool functionality in use by BPMN modelers. We proceed as follows. We review relevant literature about modeling tools and previous findings about key grammar usage beliefs. Then, we report on our data collection efforts, before presenting the results of our study. We provide a discussion of implications of the findings, and conclude the paper with a summary of its contributions and limitations.

2. Background

2.1. Prior work

In deciding how to model a real-world domain, the decision of the type of grammar to be used for conceptual modeling is an important consideration. The offset of modeling constructs and the related grammar rules define the world view of that grammar and thus specifies the limits of what can be modeled with a given grammar [33]. The type of grammar used for conceptual modeling (e.g., data-oriented, object-oriented or process-oriented) defines the language and its grammatical rules that can be used to articulate and communicate details about the real-world domain, and thus determines the outcomes of the modeling process, i.e., the type and quality of the model produced. There is a need, consequently, to understand the modeling capabilities, and limits thereof, of a modeling grammar, and the implications these limits have on the actual usage of the grammar. This understanding is of equal importance for the developers of modeling grammars, their end users as well as the developers and adopters of modeling tools.

Aside from research that studies the final *product* of using modeling grammars, i.e., the model produced (e.g., [51,65]), recently, research has been conducted to understand the *process* of using grammars for conceptual modeling. Initial studies in this vein of research examined how well grammars, in theory, support the modeling of real-world phenomena [29] or how principles of modeling could be formulated [37]. Since then, increasingly, empirical work has been carried out to understand the process of modeling with grammars. For instance, Davies et al. [13] and Fettke [23] report on surveys on the usage of conceptual modeling in practice. Dobing and Parsons [20] report on a study of how UML is used. Zur Muehlen and Recker [72] define usage patterns of the BPMN grammar. In [54], a theoretical model was developed and empirically tested that describes important grammar usage beliefs and how they lead to continued usage of a modeling grammar. This study was the first to inform a body of knowledge about the grammar usage experience as perceived by the modeler. Finally, in [61], a study is reported on the ontological characteristics of conceptual modeling grammars that determine the important grammar usage beliefs perceived usefulness and perceived ease of use.

2.2. Grammar usage and tool functionality

The studies reviewed above have arguably advanced our knowledge about how grammars are used in the process of conceptual modeling. Still, grammars are not the only modeling artifact relevant to this process. For instance, most organizations define organization-internal modeling conventions—norms that prescribe modeling guidelines, layout conventions and other standards (e.g., [63]). In this context, some studies have been conducted of late to define best practices, for instance, for naming conventions to be used in conceptual modeling [45].

Similarly, a decision for or against the use of a conceptual modeling grammar is typically associated with investments in a modeling tool to support and enact the modeling [35]. Such investments decisions are important to the effectiveness and efficiency of modeling projects [2] and are associated with significant costs. For instance, a large Australian bank estimated that its decision to introduce and use the ARIS toolset [64] for its process modeling initiatives resulted in costs of $3.5M approximately.

Aside from being an important factor in the investment decision about process modeling, the modeling tools also impact the way modeling itself is conducted. State-of-the-art modeling tools provide a graphical editor to build or read conceptual models and complement this editor with advanced functionalities to support the act of modeling as well as the utilization of the model produced. Currently, several research efforts are underway to develop advanced tool support for conceptual modeling through advanced repositories [41], advanced visualization support [7] or with advanced collaboration features [16].

Turning to modeling tool features that are widespread in industry at current, Table 1 describes the most common tool functionality in use, based on the studies in [14,30,57,58], which discuss, to some extent, how tools are used in conjunction with grammars in modeling initiatives. Table 1 further describes briefly how the tool functionality can be used to offset grammar deficiencies or other modeling challenges.

Clearly, tool functionalities such as the ones described in Table 1 influence the way that modeling is conducted, and thereby impact both the process and outcomes of