Available online at www.sciencedirect.com
ScienceDirect
Data & Knowledge Engineering 62 (2007) 292–307
DATA & KNOWLEDGE ENGINEERING
www.elsevier.com/locate/datak

Eliciting information for product modeling using process modeling

Ghang Lee [a,*], Charles M. Eastman [b], Rafael Sacks [c]

[a] *Department of Architectural Engineering, Yonsei University, 134 Sinchon-Dong, Seodaemun-Gu, Seoul 120-749, Republic of Korea*
[b] *College of Architecture, Georgia Institute of Technology, Atlanta, GA 30332-0155, USA*
[c] *Faculty of Civil and Environmental Engineering, Technion – Israel Institute of Technology, Israel*

Received 6 June 2003; accepted 16 August 2006
Available online 18 September 2006

Abstract

A product model is a formal and structured definition of product information. The most common procedure for defining a product data model is to first describe the business and/or engineering process in a formal process model, then to create a product data model based on the process model. However, there is a logical gap between process modeling and product modeling methods in the current ISO 10303 standard product modeling process. We propose a new formal approach, called Process to Product Modeling (PPM) in which process and product modeling can be logically linked. This paper focuses on the semantics and syntax for the Requirements Collection and Modeling (RCM) of PPM. The new RCM method aims to model heterogeneous business processes and their information flows. The major difference between the proposed method and traditional requirements collection methods is that it provides a theoretical linkage to integrate process models and a set of specific information items used in them. This theoretical linkage enables modelers to capture the contents, scope, granularity, and semantics of information used in activities, which are depicted in process models. A PPM method, called Georgia Tech Process to Product Modeling (GTPPM), has been developed and was deployed for the North American Precast Concrete Software Consortium; examples from this effort are presented. Experience to date indicates that the new RCM method and the GTPPM tool hold the potential to improve and expedite elicitation of information for product model development.

Keywords: Product model; Process model; Information flow; CASE; STEP; GTPPM

1. Introduction

A *product model*[1] is a formally structured schema carrying product instance information that is generated, deleted, and modified through the lifecycle of a product. A product model is a standard medium to share and

E-mail addresses: glee@yonsei.ac.kr (G. Lee), chuck.eastman@arch.gatech.edu (C.M. Eastman), cvsacks@techunix.technion.ac.il (R. Sacks).
[1] In this paper, the terms 'a product model' and 'a product data model' are used interchangeably.

0169-023X/$ - see front matter
doi:10.1016/j.datak.2006.08.005

exchange product information electronically among heterogeneous systems within an organization, or more widely, within/across industries. Examples include ISO 10303 STEP: STandard for the Exchange of Product model data [1], CIS/2 – a US standard product model for the steel construction domain [2], and IFC – an international standard product model for the architecture, engineering, and construction domain [3]. (*Note*: This paper uses many acronyms including the STEP terms. The acronyms are summarized in Glossary at the end of this paper.)

While significant effort has already been applied to the development of product models, many engineering and production domains are still developing their IT infrastructure and have not yet developed corresponding product models. Product models are live and not static, requiring updating as new technologies and concepts are integrated into a manufacturing or design domain. The need to develop product models will continue for the foreseeable future.

The current standard method employed in most current product modeling efforts is based on the ISO-10303 STEP languages and methods [4,5]. Like other data modeling methods, the STEP standard procedure begins by collecting requirements. It defines the scope and processes to be supported by developing a process model of the universe of discourse, (called an *Applications Activity Model* (*AAM*)), using IDEFØ. The AAM is basically a process model [6] and IDEFØ is a standard process (or function) modeling language for ISO-10303 [4,7]. Although not described in these terms, it is part of the *requirements collection* phase in data modeling [8]. Later, the AAM can be used in defining the scope of an *Application Requirement (or Reference) Model (ARM)* in various graphic and non-graphic data modeling languages IDEF1x, NIAM [9], EXPRESS-G, or EXPRESS. EXPRESS is the textual product modeling language universally used in such efforts [10–12]. An Application Requirements Model (ARM) is basically a data model for products. It is a "detailed specification of the data objects (entities and attributes) and the relationships between them that are required to support the activities within the scope of the industry application [13]". This is called the *logical (schematic) data modeling* phase. The ARM is then refined, elaborated, and integrated with other standard product models. The integrated model is called an *Application Interpreted Model (AIM)*.

The data modeling process as described is commonly undertaken by modeling experts and domain experts [4,12]. It is an iterative process, in which the participants aim to reach consensus on the correct models (first process and then product). Specification and approval of a product model typically takes three to five years to complete the full cycle. Throughout later stages, applications within the domain are engaged and translators to/from the product model are developed. Initial interfaces typically involve file-based exchange, with database implementations following later. The STEP method using IDEFØ for process modeling has been adopted by many organizations such as US Air Force, IAI [14], and a number of projects carried out under the auspices of the European Union [15,16].

However, the ISO STEP product modeling method suffers from a number of drawbacks:

1. It relies exclusively on *iterative human review* for validation. Human review is necessary for capturing semantic fallacies that cannot be captured by logical validation methods, but it is time consuming and error-prone. Also, there are *no formal rules governing the consistency of information flows* in a model, and the limitation described below in (4) prohibits implementation of automatically processed logical rules for checking consistency between the process and product models, which are necessary for effective information level process modeling.
2. In almost all industry-wide product modeling efforts, IDEFØ models are built as single generic models to represent idealized industry-wide processes, defined by consensus among multiple stakeholders [4,17]. The single communal models are inevitably *high-level*, lacking detailed activities and information items used by the activities. In this approach, each individual company must plan integration of its systems with the product model separately from the communal activities. There is no means to include local variations in the modeling effort, or to validate that the product model developed supports current or anticipated corporate processes.
3. Current product data models are defined as *static structures*, more as archives of data than to support strategic workflow processes. The developmental and evolutionary aspects of product development and production planning are not well supported [18,19]. If product models are to truly support process re-engineering and integration, closer linkage with the workflow characterization of a product domain is required.