Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 112 (2017) 544–553

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France

MOT Knowledge Model Integration Rules for Knowledge Warehousing

Rim Ayadi[a,*], Yasser Hachaichi[a,b], Jamel Feki[a,c]

[a] *Multimedia InfoRmation Systems and Advanced Computing Laboratory, University of Sfax, Sfax, Tunisia*
[b] *Department of Computer Science and Quantitative Methods, Higher Institute of Business Administration of Sfax*
[c] *Faculty of Computing and IT, University of Jeddah, Jeddah, Saudi Arabia*

Abstract

A knowledge warehousing process aims to build an intelligent decision support system. It collects, homogenizes, integrates and stores knowledge for a decision-making process. In this paper, we are interested in knowledge integration. More accurately, we propose an integration process for knowledge homogenized/modeled according to the MOT (Modeling with Object Types) knowledge model. This integration process consists of three ordered steps based on the type of schemas to integrate and their similarity. For this process, we define five integration rules based on semantic relationships between elements of MOT knowledge models, and then we develop an algorithm using these integration rules.

Keywords: Knowledge warehouse; Knowledge integration; MOT Knowledge; Semantic relationships; Integration rules.

1. Introduction

The knowledge management process, intended to create and exchange knowledge between individuals inside and outside an organization[1,2], has been developed significantly since knowledge is a powerful means of success for organizations. This knowledge is either *explicit* presented in a tangible form, or *tacit* made up of acquired skills of experts[3]. However, the obtained knowledge is a bit organized, heterogeneous and scattered in various systems. It thus looks vital to gather, homogenize, integrate and organize this great mass of knowledge for an effective use by decision makers. For this purpose, some research works introduced the concept of *knowledge warehouse* (KW)[4,5,6,7].

Indeed, through a KW, our aim is to satisfy the decision makers needs better decisional performances, by offering an integrated global vision of the diversified knowledge of the organization. This knowledge is initially expressed under different knowledge models (e.g., association rules, decision trees, neural networks, clusters[8])[9]. Inevitably, a KW aims to solve this heterogeneity. To reach this end, we opted for the semi-formal graphical language for knowledge modeling called MOT (Modeling with Object Types)[10] as a unified/pivot language to homogenize knowledge

* Corresponding author. Tel.: +216 95 114408.
E-mail address: rim.ayadi@yahoo.fr

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of KES International
10.1016/j.procs.2017.08.164

models[7], thus facilitating their integration. MOT is a language based on a graphical formalism, which makes it easy to use by IT persons as well as by decision-makers.

In MOT, the pieces of knowledge units (KU) are represented with graphical symbols, which distinguish their abstraction levels, their types and their semantics[10,11]. In fact, each KU is labeled by its name and it can be an abstract knowledge unit AKU {*Concept (class of objects), Procedure (class of actions) or Principle (class of statements, conditions, properties, agent/actor or rules)*} or their corresponding factual knowledge unit FKU respectively called {*Example, Trace or Statement*}. These pieces of KU are connected by links that can be of type {*Instantiation (I), Composition/multiple Composition (C/C*), Specialization (S), Precedence (P), Input/Product (I/P), Regulation (R) or Application (A)*}, each having an appropriate semantics[10,11] and complies with integrity rules described in literature[10,12]. These rules define the valid relationships between the different KU.

In our previous work we proposed a set of rules to transform into MOT knowledge models extracted from data sources using three data mining techniques namely *decision trees DT*, *association rules AR* and *clustering CL*. These rules are based on structural correspondences between, on the one hand, each source Meta-model (MM) and, on the other hand the MOT target MM that we have defined and extended with the AKU of type *PrincipleProcedure* that results from a multiple inheritance of the types *Principle* and *Procedure*[7].

In this paper, we aim to unify pieces of knowledge transformed into MOT and integrate them in a global repository. To carry out this integration task, we need to identify the semantic relationships between KU of MOT knowledge models in order to define the integration rules for the unification of MOT knowledge models.

We organize the remainder of this paper as follows: Section 2 overviews the state of the art about strategies of schemas integration; it also introduces our proposed approach for a knowledge integration process. Section 3 presents the various semantic relationships that we can identify between the KU of MOT models. In Section 4, we define our set of five rules for the integration of MOT models. In Section 5, we develop our integration algorithm and we illustrate it with an example. Finally, Section 6 concludes this paper, states our on-going research activities in this context, and projects our future concerns.

2. Integration strategies: An overview and proposed process

The literature classifies the schema integration strategies into two families known as *Binary* and *n-ary*[13,14,15].

The family of *Binary* strategies allows the integration of the schemas in pairs. It can follow a *Ladder* or a *Balanced* strategy. *Ladder* integrates at each step a new schema with an intermediate result schema; *Balanced* divides the schemas into pairs at the beginning and integrates them symmetrically. Then, it integrates in pairs the intermediate result schemas. The *n-ary* strategy allows the integration of n ($n > 2$) schemas simultaneously. An *n-ary* strategy is *one shot* when the n schemas are integrated in a single step; it is *iterative* otherwise.

Binary strategies perform comparison tasks simply at all integration steps and minimize the number of comparisons between the schemas concepts[13]. Most works follow a *binary* strategy since these strategies simplify the complexity of the integration process essentially when the number of schemas to be integrated increases[13,14,16]. Moreover, an intermediate integration step solves conflicts that may occur between schemas. For this reason, a great importance is associated with an already existing partially integrated schema[13]. In addition, in *binary* integration the domain expert can select the order of the schemas to integrate.

As for the *n-ary* integration strategies, the number of iterations is reduced to a minimum. Moreover, a significant quantity of semantic analyses can be carried out before fusion, which avoids the need for a thorough analysis of the integrated result schema[13]. The greatest difficulty of *n-ary* integration strategies is the analysis and merging of participating schemas while still separating and keeping track of the changes to each of their elements, so that mappings between participating and integrated schemas can be established later[14]. Furthermore, there is not a flexible and powerful framework to undertake these *n-ary* integration strategies[14].

Considering the advantages of the *binary* strategy, we opt for a *binary* integration process combining its two alternative strategies *Ladder* and *Balanced*. More precisely, we choose the *Ladder* integration strategy when the MOT schemas to be integrated result initially from the same knowledge model and we resort to the *Balanced* strategy when MOT schemas are of distinct types.

These two strategies allow having a single unified schema[13] at each integration step. However, our integration process produces two schemas; i.e., two MOT knowledge models. In fact, if we find semantic relationships between