Computer Languages, Systems & Structures 44 (2015) 299–318

Contents lists available at ScienceDirect

Computer Languages, Systems & Structures

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

Concepts and evaluation of the extended entity-relationship approach to database design in a multi-paradigm information system modeling tool

Vladimir Dimitrieski *, Milan Čeliković, Slavica Aleksić, Sonja Ristić, Abdalla Alargt, Ivan Luković

University of Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia

ARTICLE INFO

Article history:
Received 15 December 2014
Received in revised form
5 August 2015
Accepted 31 August 2015
Available online 8 September 2015

Keywords:
Entity-relationship
Domain-specific language
Model transformation
Information system design
Evaluation study

ABSTRACT

Different approaches to information system (IS) development are based on different data models. The selection of a data model for conceptual design, among other things, depends on the problem domain, the knowledge, and the personal preferences of an IS designer. In some situations, a simultaneous usage of different approaches to the conceptual database design and IS development may lead to the most appropriate solutions. In our previous research we have developed a tool that provides an evolutive and incremental approach to IS development, which is based on the form type data model. The approaches based on the Extended Entity-Relationship (EER) and class data models are broadly accepted throughout the community of IS designers. In order to support the simultaneous usage of approaches based on the form type, EER and class data models, we have developed the Multi-Paradigm Information System Modeling Tool (MIST). In this paper, we present a part of our MIST tool that supports EER approach to a database design. MIST components currently provide a formal specification of an EER database schema specification and its transformation into the relational data model, or the class model. Also, MIST allows generation of Structured Query Language code for a database creation and procedural code for implementing database constraints. In addition, Java code that stores and processes data from the database, may be generated from the class model. In this paper, we present the evaluation study of the MIST EER domain-specific language. Users' perceptions of language quality characteristics are used for the evaluation.

1. Introduction

In the last few decades, a number of information system (IS) development approaches have emerged. Some of the methods that are still in use are based on: the Entity-Relationship (ER) data model proposed by Chen [1] with its further extensions (Extended ER, EER), the relational data model [2], the form type (FT) meta-model [3], and the object-oriented model [4].

* Corresponding author.
E-mail addresses: dimitrieski@uns.ac.rs (V. Dimitrieski), milancel@uns.ac.rs (M. Čeliković), slavica@uns.ac.rs (S. Aleksić), sdristic@uns.ac.rs (S. Ristić), abdalla@uns.ac.rs (A. Alargt), ivan@uns.ac.rs (I. Luković).

http://dx.doi.org/10.1016/j.cl.2015.08.011
1477-8424/

In the context of the Model-Driven Software Development (MDSD), EER and relational data models may be seen as EER and relational meta-models respectively. A database schema is expressed by means of the concepts of a selected meta-model. According to the MDSD terminology, such database schema is called a database model and it should conform to the appropriate meta-model. An EER database model conforms to the EER meta-model, while a relational database model conforms to the relational meta-model. In this paper, we use the MDSD terminology based on the four-layered architecture proposed by OMG [5]. In Fig. 1, we present notions used in this paper by gray rectangles. The white rectangles represent the corresponding notions from IS design terminology. Shapes filled with diagonal stripes are notions with the same name in both MDSD and IS design terminologies. Hereinafter, the notion "EER approach" is used to express the approach to an IS or a database schema modeling based on the EER data model, while the notion "FT approach" is used for the approach to an IS or a database schema modeling based on the FT meta-model.

Throughout our previous research [3,6–8], we have developed a tool named IIS*Studio that provides a FT approach to evolutive and incremental IS development. The approach is purely platform independent and strictly differentiates between the specification of a system and its implementation on a particular platform. A detailed description of the FT data model and FT approach implemented in IIS*Studio may be found in [3,7,8]. The EER approach is broadly accepted in the community of IS designers. Therefore, we have decided to extend the IIS*Studio functionalities in order to support the EER approach, too. By this, we have developed: (i) a DSL for creation of EER database schema specifications, named EERDSL; and (ii) Eclipse-based tool that supports the FT and EER approaches, both providing conceptual database schema modeling. The tool is named Multi-Paradigm Information System Modeling Tool (MIST). The MIST development is based on the MDSD paradigm and Eclipse Modeling Project (EMP) tools. In MIST, FT and EER approaches may be used simultaneously. For both FT and EER database models, MIST provides transformations into a relational data model. In our previous research on database reengineering approaches [9], we have developed a transformation from a relational database model to a FT model, named relational-to-form type transformation (R2FT). In MIST, we use R2FT as a link in the chain of transformations aimed to transform an EER database model into a corresponding FT model via a relational database model. Our primary idea was to use MIST for educational purposes, such as learning about: (i) EER data model concepts and developing a database schema specification at the conceptual level; (ii) transformations of EER to relational database schema specifications; (iii) transformations of EER to class models; and (iv) MDSD approach, by means of the EER approach the students are familiar with, since it is extensively taught in the database courses. Integrated with IIS*Studio, MIST can be successfully used in the design and generation process of a database and application software of an IS.

In the paper, we present the architecture of MIST. It comprises several components that support conceptual modeling based on the FT and EER approaches, as well as code generation. The focus of this paper is set on the components supporting EER approach. The description of other components may be found in [3,6–9]. Since a main goal of developing MIST is to deploy it both in education and practice, we have performed evaluations of the tool so as to recognize its strengths and shortcomings. In this paper, we present the findings of the following two types of evaluation:

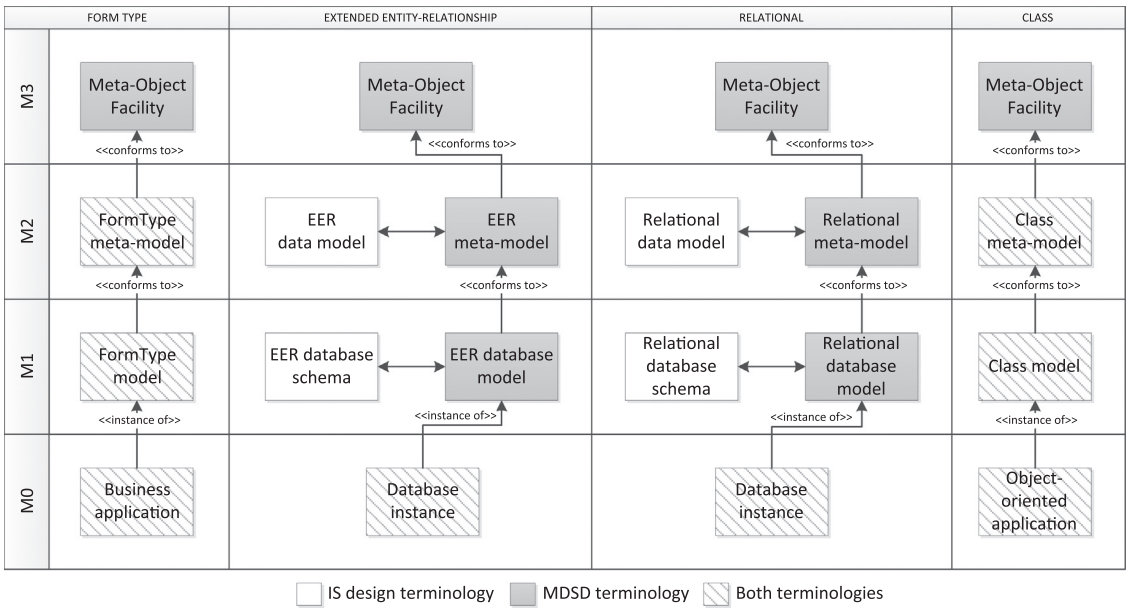

Fig. 1. Terminology used in this paper alongside with the notions from the IS design domain.