Criteria for Description of MDE Artifacts

Fábio P. Basso[1], Toacy C. Oliveira[1], Cláudia M. L. Werner[1], Valdemar V. Graciano Neto[2,3,4], Flavio Oquendo[4], Elisa Yumi Nakagawa[3]

[1]PESC - Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil

[2]INF - Federal University of Goiás, Goiânia, GO, Brazil

[3]ICMC - University of São Paulo (USP), São Carlos, SP, Brazil

[4]IRISA-UMR CNRS - Université de Bretagne-Sud, Vannes, France

{fabiopbasso,toacy,werner}@cos.ufrj.br, valdemarneto@inf.ufg.br

flavio.oquendo@irisa.fr, elisa@icmc.usp.br

Abstract. *The large-scale collaboration in Model Driven Engineering (MDE) demands the use of one or more repositories such as GEMOC, ReMoDD and SEMAT, which share artifacts (e.g., models, meta-models, transformations, etc) for global reuse scenarios. While recent researches motivated the need of data for decision making from the perspective of Software Ecosystems (SECOs), the cutting edge technologies on this topic merely classify these artifacts. Likewise, it is important the definition of criteria to represent this qualified data with enough semantic potential to support reuse of MDE Artifacts globally, which is missed in the literature of the area. This paper proposes some criteria, thus a small contribution to help on the establishment of a future SECO for MDE.*

1. Introduction

Model Driven Engineering (MDE) is an approach that considers models as first class citizen in software development [Whittle et al. 2015]. Typical artifacts in MDE (termed as **MDE Artifacts**) often include models, metamodels, model transformations, model managers, and domain-specific models. MDE as Service (MDEaaS) is a novel approach for MDE that seeks to provide access to MDE through services [Basso et al. 2015]. Such artifacts are supposed to be stored and discovered from repositories or Knowledge Bases (KB) to be downloaded and integrated into software projects, as illustrates Figure 1.

In Phase 1, this approach can foster reuse of artifacts previously produced through repositories, reducing cost and time-to-market to integrate them in appropriate representations in Phase 3 [Mussbacher et al. 2014]. Repositories, to be effective, need a sort of semantic information (so-called *qualified data*) associated with such artifacts. These data shall describe applicability, guide the reuse of artifacts and allow the comparison between different features from artifacts in Phase 2, thus requiring well structured and qualified data. However, the literature does not properly offer solutions for this phase.

In a previous work, we claimed that the execution of software engineering services for reuse of MDE Artifacts in global scale can benefit from a perspective of Software Ecosystem (SECO) [Basso et al. 2015], whose represented

Figure 1. A scenario for qualified descriptive data of MDE Artifacts.

assets are often analyzed under a three-dimension perspective (technical, business, and social) [Santos and Werner 2012]. Likewise, a first step towards the advent of a SECO for MDE is to meet reuse opportunities from repositories such as SEMAT [Jacobson et al. 2012], GEMOC [Combemale et al. 2014] and ReMoDD [France et al. 2007]. This requires the representation of assets with qualified data that describes and supports a three-dimension perspective of MDE Artifacts. However, assets are currently specified following an ad-hoc strategy, which hampers the execution of the second phase of Figure 1. Thus, our contribution is a set of criteria for representation of descriptive data for these assets.

This work is organized as follows: Section 2 provides background in repositories for MDE Artifacts; Section 3 proposes a criteria for representation of descriptive information associated with MDE Artifacts; Section 4 presents some insights for future works and Section 5 summarizes our conclusions.

2. Repositories for MDE Artifacts

MDE is often adopted to engineer domain-specific solutions. As such, many artifacts produced to comply with a subset of requirements in one project could be adopted again to develop other product which integrate the portfolio of a company. However, few is discussed about the quality of descriptive information to be associated with the shared artifacts, which is critical for decision making for one asset or other to be introduced in specific contexts [Whittle et al. 2015]. Figure 1 illustrates how this could be achieved considering the current availability of repositories in a possible implementation of a MDEaaS approach. Hence, artifacts could be searched, analyzed and integrated in new MDE projects.

Currently, existing work proposes the adoption of a unique underlying infrastructure (a unique KB) that fosters reuse of MDE Artifacts in Phase 1. Much of the effort in repositories is dedicated to tool support, focusing in classification [Lúcio et al. 2014]. REMODD is a repository for MDE that can be used to tackle the problem of reusing and sharing MDE Artifacts [France et al. 2007]. An initiative to accomplish a KB named SEMAT focuses on reusable methods [Jacobson et al. 2012]. SEMAT adopts Essence as a core representation language, which is useful to represent methods but limited to represent data from MDE technicalities. GEMOC [Combemale et al. 2014] is another proposal in this direction.

Instead this simplest approach, the execution of MDEaaS could consider the reuse of artifacts distributed in these several repositories. This needs a common representation language and the uniformity of information [Basso et al. 2015]. However, although some representation languages are available for representation of data

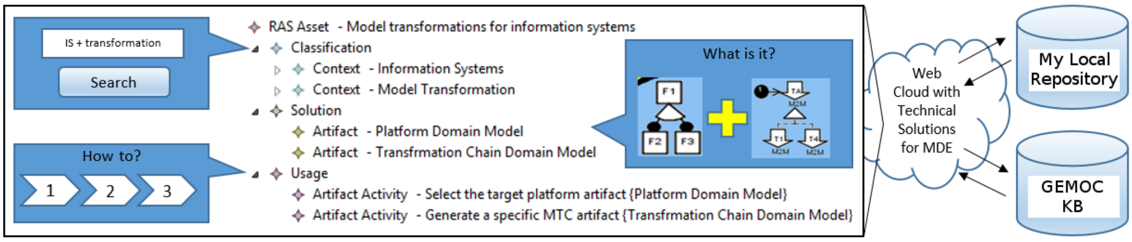

Figure 2. Information represented in a reusable asset conforms to RAS.

independently from repository providers [Basso et al. 2016], there is no criteria for representation of MDE assets. Besides, there is no requirements for data in support for SECO perspectives in MDEaaS. This limitation is tackled in this paper.

3. Criteria for Qualified Descriptive Data

Figure 2 exemplifies data associated with two MDE Artifacts in an asset. It is based on the Reusable Asset Specification (RAS) [Basso et al. 2016], a standard from the Object Management Group (OMG) for externalize data associated with artifacts from reuse repositories, i.e., making an asset independent from a repository vendor (e.g., My Repository or GEMOC KB). An asset conforms to RAS allow the use of some structures for descriptive data: 1) *Classification*, allowing the automatic cataloging in repositories; 2) *Solution*, allowing the detailing of data from artifacts such as type, variability points and semantics; and 3) *Usage*, allowing the representation of instructions for reuse after acquisition/download.

Motivation: Instead of the imposition of use of a unique KB, we found relevant the use of them all through a common representation. Likewise, we need to reach criteria for representation of information for assets independently from a repository [Basso et al. 2015]. Although RAS plays a similar role as layers in a network protocol that are processed in levels of abstraction, it is also important use criteria for representation of data associated with MDE Artifacts. Thus, we present criteria for data used in Phase 2 considering technical, business, and social needs.

Criteria formulation: It was conducted an analytical study of representativeness of two asset specification languages [Basso et al. 2016]. We also looked in the literature based in a structured keyword search for recommendations to descriptive data, but we missed focused papers that could be used. Then, we conducted a multivocal literature review [Ogawa and Malen 1991], which is mainly characterized by the inclusion of material not available in search engines. The following source of information is investigated in a multivocal research protocol: a) searched in books about design patterns and empirical software engineering; b) searched in papers on management of model-based operations; and c) looked for calls for contributions (tool demo) from conferences related with the MDE (MODELS, ECMFA, SPLC, GPCE, SLE, etc.) and in more general conferences (ASE, ICSE, OOPSLA).

Criteria for the descriptive-level of MDE Artifacts:

- **Information about the context.** It is important to represent information that answer some questions recommended in book for design patterns [Gamma et al. 1995]: a) What is the solution? b) How to use it? c) Who can use it? d) Known uses? e) Which are the known incompatibilities? f) Which is the software license and its dependencies? g) Examples, etc.

- **Information about the application.** Questions recommended by empirical engineering [Whittle et al. 2015, Mussbacher et al. 2014] are important to establish comparisons between competitor MDE Artifacts such as: a) Which are the minimum requirements to use an MDE Artifact including teams configuration, members skills (social and technical) which the artifact have been used successfully, the required knowledge from end-users about domain specific languages, with which software process and so on? b) Which are the pre and post-conditions to use it? c) Which are the evidences that support the solution and in which boundary conditions? d) Which is the learning curve and the Return on Investment (ROI)? and e) Feedback from users.
- **Quality for the descriptive information.** The following questions should be considered when representing descriptive data from MDE Artifacts: a) Is the information for catalog based in standard taxonomy for searching technical solutions? b) Is this standard taxonomy represented in a KB as data dictionary, thus describing what means the contexts used in catalog information? c) Is clear the instructive information such as how to integrate and use artifacts in contexts?

Criteria for descriptive and technical-level data:

- **Structure for the descriptive information.** The following information should be structured to enable analysis of MDE Artifacts: a) Abstractions for types of artifacts produced and consumed in software processes phases by a tool in the context of MDE; b) Abstractions that makes clear when an MDE Artifact needs adaptation before introduction in a target context; c) Textual information for decision making (business, technical and social perspectives), which could be based on structures for design patterns, so one can decide the option that best meet a specific need in a software development context.
- **Link to technical information.** Descriptive information itself is useless, which means that technical information should also be considered such as semantics to: a) The target platform associated with transformations for model-to-text/model-to-code/code-to-model; b) The type of the model transformation [Lúcio et al. 2014]; c) The meta-models and the meta-model framework [Combemale et al. 2014], as well as the design languages associated with each model transformation, such as UML Profiles, or Graphical DSL, or Textual DSL; e) Model transformation components, bindings and parameter matching; and f) Variability points [Basso et al. 2016].

4. Ongoing and Future Works

Figure 2 shows the structures for representation data for MDE Artifacts, which is interesting as a common language, but limited when applying the proposed criteria. In order to enable the implementation of the scenario illustrated in Figure 1, future works shall further investigate the following aspects: a) New abstractions for a *Classification* structure, which allows the storage and search of MDE Artifacts in a KB with data about context and application (Phase 1); b) Structures for design patterns, used for representation of data to support decision making (Phase 2); and c) Abstractions for technicalities in settings, represented as instances of *Artifact* for MDE specificity (e.g., chains of components) connected with a structure called *Usage* (Phase 3).

5. Final Remarks

This paper address a set of criteria with requirements for representation of descriptive data of MDE Artifacts. These artifacts are currently available in repositories for MDE allowing (i) support their use through services, (ii) facilitate their discovery and understanding of such artifacts, and (iii) foster their systematic reuse. To be effective, it is necessary to represent qualified descriptive data, which add semantic to such artifacts, thus promoting their adoption in new MDE software development projects. So far, it is not clear what must be represented in this level of abstraction considering a SECO perspective. Thus, through presented criteria and work done, this perspective can now be introduced in MDE as Service based in a common format for representation of assets and in recommendations for descriptive data.

References

Basso, F. P., Oliveira, T. C., and Werner, C. M. L. (2016). Analysis of asset specification languages for representation of descriptive data from mde artifacts (to appear). In *Conference on ENTERprise Information Systems, October 5-7*.

Basso, F. P., Werner, C. M. L., and Oliveira, T. C. (2015). A summary of challenges for "mde as service". WDES'15, pages 85–88, Belo Horizonte-MG, Brazil.

Combemale, B., Deantoni, J., Baudry, B., France, R., Jézéquel, J.-M., and Gray, J. (2014). Globalizing modeling languages. *IEEE Computer*, 47(6):68–71.

France, R., Bieman, J., and Cheng, B. (2007). Repository for model driven development (remodd). 4364 LNCS, pages 311–317.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns, Elements of Reusable Object-Oriented Software*. Addison-Wesley.

Jacobson, I., Ng, P., McMahon, P. E., Spence, I., and Lidman, S. (2012). The essence of software engineering: The semat kernel. a thinking framework in the form of an actionable kernel. *ACMQueue. Development 9. Networks*, 10(10):1–12.

Lúcio, L., Amrani, M., Dingel, J., Lambers, L., Salay, R., Selim, G. M., Syriani, E., and Wimmer, M. (2014). Model transformation intents and their properties. *Software & Systems Modeling*, pages 1–38.

Mussbacher, G., Amyot, D., Breu, R., Bruel, J.-M., Cheng, B. H., Collet, P., Combemale, B., France, R. B., Heldal, R., Hill, J., Kienzle, J., Schöttle, M., Steimann, F., Stikkolorum, D., and Whittle, J. (2014). The relevance of model-driven engineering thirty years from now. In *MODELS*, pages 183–200.

Ogawa, R. T. and Malen, B. (1991). Towards rigor in reviews of multivocal literatures: Applying the exploratory case study method. *Review of Educational Research*, 61(3):265–286.

Santos, R. P. d. and Werner, C. (2012). Reuseecos: An approach to support global software development through software ecosystems. ICGSEW, pages 60–65, Washington, USA. IEEE.

Whittle, J., Hutchinson, J., Rouncefield, M., Håkan, B., and Rogardt, H. (2015). A taxonomy of tool-related issues affecting the adoption of model-driven engineering. *Software & Systems Modeling*, pages 1–19.