Domain-Specific Conceptual Modeling: An instrument for increasing productivity in system development

Martin Paczona [1], Heinrich C. Mayr [2] und Guenter Prochart[3]

Keywords: domain-specific conceptual modeling, DSML, testbed model, evaluation, productivity, modeling tool

Abstract: This EMISA24 ”Current Research Talk” is based on the authors’ paper [PMP24], recently published in the journal Data and Knowledge Engineering. We address the question of whether and how domain-specific modeling can increase productivity in the development of technical systems in an industrial setting. For, managers’ decisions are ultimately based on whether or not the use of a new method pays off. The research was intertwined with a project in which we collaborated with a company to develop a domain-specific modeling method and tool for designing and implementing testbeds for electric vehicles.

Summary

Our focus is on productivity in the process of technical system development, where productivity is roughly defined as the relationship between production output and production input such as labor, capital, and environment. The research questions addressed are:

- R1: What are the productivity factors in technical system development?
- R2: How are these factors affected by using a DSMM for development?
- R3: What effort should one expect to expend in developing a DSMM?

We tackled these questions in accompanying research for a project in which we developed a DSMM for Electric Vehicle Testbeds (EVT) [PM19; PMP20] in cooperation with an Austrian enterprise. In the paper we (1) introduce a taxonomy of productivity factors in system development, (2) examine how these factors are influenced with domain-specific modeling, and (3) apply them to the development of our method itself.

This is a work in the field of Design Science. We have (1, ”*Rigor Cycle*”) reviewed the existing knowledge in the research area, (2, ”*Relevance Cycle*”) developed an artifact and (3, ”*Evaluation Cycle*”) assessed our DSMM in case studies addressing the following aspects: (1) Appropriateness of the DSML, (2) Tool introduction and first exercises, (3) Remodeling graphical testbed drawings from previous projects, (4) Generating models from customer orders, (5) Applying the DSMM in real projects, (6) Development phase, and (7) Usage in support.

1 Alpen-Adria-Universität Klagenfurt , 9020 Klagenfurt , Austria,
mapaczona@edu.aau.at, https://orcid.org/0000-0002-6330-5974
2 Alpen-Adria-Universität Klagenfurt , 9020 Klagenfurt , Austria,
heinrich.mayr@aau.at, https://orcid.org/0000-0001-5770-8091
3 Virtual Vehicle Research GmbH , 8010 Graz , Austria, guenter.prochart@v2c2.at

doi:10.18420/EMISA2024_04

The productivity drivers in development processes, that we have identified, are (together with the results gained from the evaluation) the following:

1. **Error prevention and correctness support**: Errors in the early development phases lead to massive costs if they are not prevented or detected in time. DSMMs provide tool-supported use of domain-specific conceptualizations, guidelines, style guides, templates, constraints etc. Evaluation: Over 50% of the analyzed testbeds had one or more constraint violations that could have been detected by the implemented constraint checks. I.e., our approach reduces the number of errors in testbed design.
2. **Methods and tools tailored to design & development**: Conceptualizations that correspond to the tasks and procedures in a domain can be learned and applied more easily. Modern DSMMs provide such conceptualizations as well as appropriate notations. Evaluation: For quantifying acceptance we analyzed the modeling effort, the number of delete actions during modeling and the understandability. It turned out that the DSML is easy to understand, and that development and modification effort is reduced. The analysis of the modeling protocols of inexperienced users showed that they proceeded very purposefully and only had to make a few corrections.
3. **Reuse of artifacts**: Models are artifacts; repositories for their reuse offer at least a search function. More advanced features support multiple metamodels and thus models formed in different languages, and their consistent composition or the reuse of templates. Evaluation: Our repository contains over 100 models, allows the import and export of models and other artifacts, e.g., parts lists. Template models can be reused for new testbed designs.
4. **Automation of work units**: In design and development this is possible wherever artifacts can be generated with the help of rules from specifications or models. The key tasks in testbed development (creating the software configuration, developing circuit diagrams and specifying of control parameters) were previously performed manually and are now handled in the EVT Modeler by generators implemented in AdoScript and Python. In the development, time-consuming checks are shortened by automatic constraint checking.
5. **Design for subsequent maintainability**: Every technical system must be maintained and adapted to new requirements. Thus, easy maintainability must be planned for in design and development. Adaptability might be supported by DSML extension mechanisms.
6. **Appropriate product support**: Comprehensive and easy-to-understand system descriptions can be a great help in product support. This is also true for DSMMs. Evaluation: Our instruments help the support staff to detect design errors and provides suggestions for improvement. Testbed-specific software configurations and control parameters can be re-generated automatically without the need for manual tuning.

Literatur

[PM19] Paczona, M.; Mayr, H. C.: Model-Driven Mechatronic System Development. In: Proc. 2019 IEEE 15th Int. Conf. on Automation Science and Engineering (CASE). IEEE, Vancouver, BC, Canada, S. 1730–1736, 2019.

[PMP20] Paczona, M.; Mayr, H. C.; Prochart, G.: Model-Based Generation of Software Configurations in Mechatr. Systems. In: Modellierung 2020. LNI P-302, 2020.

[PMP24] Paczona, M.; Mayr, H. C.; Prochart, G.: Increase development productivity by domain-specific conceptual modeling. Data & Knowledge Engin. 150/, 2024.