Content-validation questionnaire of a meta-model to ease the learning of data visualization concepts

Andrea Vázquez-Ingelmo [1], Alicia García-Holgado [1], Francisco José García-Peñalvo [1], Roberto Therón [1] and Ricardo Colomo-Palacios [2]

[1] *GRIAL Research Group, University of Salamanca, Paseo de Canalejas 169, Salamanca, Spain*
[2] *Østfold University College, Halden, Norway*

Abstract

Data visualizations and dashboards are powerful means to convey information to large audiences. However, the design and understanding of these tools are not straightforward because several factors are involved. It is essential to rely on theoretical frameworks to design and implement data visualizations for these reasons. In this context, we propose a meta-model to identify and arrange the main characteristics and elements of data visualizations and dashboards. The proposed meta-model provides a powerful artifact to generate information visualizations and dashboards automatically, but also a learning resource to understand how data visualizations elements interact and influence each other. However, it is necessary to validate this artifact to ensure its quality and usefulness. In this paper, we present a work-in-progress or a quality assessment and content validation of the me-ta-model to seek weaknesses and tackle them in subsequent iterations.

Keywords

Information Dashboards, Data Visualization, Learning Resource, Content Validation.

1. Introduction

Visual explanations are everywhere: they convey complex information, raise attention over target topics, improve the understandability of particular domains, etc. [1]. They can take the form of infographics, simple graphs, or even elaborated information visualizations.

These tools are also powerful because they let users visually perceive information to generate knowledge. However, the complexity involved in this domain can hamper the understanding of the displayed data.

Data visualizations are composed of different visual elements, including shapes, visual encodings (i.e., visual characteristics like color, size, position, etc.), and visual aids (i.e., legends, axes, etc.), but also abstract concepts like scales, data domain, data operations, among others.

If these concepts are not fully understood, they can lead to improper designs (on the developer's side) and to wrong conclusions (on the audience's side).

It is essential to consider all the concepts and factors involved in the data visualization domain to provide adequately designed and reliable visualization methods.

These factors can be considered through domain experts (i.e., information visualization experts) who also know the visualization's data do-main and can provide a well-designed product through its expertise.

However, it isn't easy to have this expertise or domain knowledge levels for every practitioner who uses information visualizations to convey information. For all these reasons, it is crucial to deeply understand all the elements that compose data visualizations and how they relate to and in-fluence each other.

Learning Analytics Summer Institute Spain (LASI Spain) 2022, June 20–21, 2022, Salamanca, Spain
EMAIL: andreavazquez@usal.es (A. 1); aliciagh@usal.es (A. 2); fgarcia@usal.es (A. 3); theron@usal.es (A.4); ricardo.colomo-palacios@hiof.no (A. 5)
© 2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3238/paper1.pdf

The abstraction of these elements and their relationships can provide a framework to improve the knowledge about the most primitive aspects of data visualizations, no matter if they are tangible (understood as the elements that are directly displayed through data visualizations, such as shapes, encodings, layouts, etc.) or conceptual (data domains, data transformations, user characteristics, etc.).

To achieve this, paradigms like the model-driven architecture [2] provides guidelines to develop a meta-model that captures the most relevant factors involved in designing data visualizations and information dashboards. We have presented a dashboard meta-model in previous works and improved it through domain engineering. The prior version of the meta-model will be described in the methodology section.

One of the main benefits of relying on a meta-model is that it can be used as a conceptual map and educational tool to guide the design of data visualizations, but also as an artifact to automatically generate analytical dashboards in different domains (health [3], education [4, 5], employment [6], etc.). However, it is crucial to validate the content of this resource to check if the represented entities are relevant, coherent, and understandable.

In this paper, we provide a proposal to validate the content of the dashboard. By validating the content of the meta-model, it is possible to identify potential limitations and drawbacks of the dashboard domain representation and address them before using this artifact to instantiate real-world dashboards.

The rest of this paper is organized as follows. Section 2 describes previous works related to meta-model validations. Section 3 describes the methods employed to carry out the meta-model, the automatic generation of dashboards, and the experts' validation questionnaire. Finally, section 4 discusses the results, and section 5 offers the conclusions derived from this work.

2. Background

A meta-model is considered of quality if it contains the required elements to instantiate a model that adequately represents the elements in the domain and is technically built using solid principles. The first concern is related to meta-model validation, which is necessary to ensure that we are building the right meta-model [7]. It is important to validate meta-models concerning the specifications of the domain or with the help of domain experts who can provide meaningful examples of correct and incorrect uses of the Domain-Specific Modelling Languages (DSML) [8]. This validation should be combined with a set of quality attributes.

There are few quality models or frameworks to measure the quality of a meta-model [9, 10]. On the one hand, Basciani, et al. [11] identify four quality assessment approaches focused on meta-models. QM4MM proposes a refinement of the ISO/IEC 9126 quality model [9]; it identifies a set of quality attributes organized in six dimensions – functionality, reliability, usability, efficiency, maintainability, and portability. Ma, et al. [10] describe a quality model that measures five quality attributes – syntactic quality, semantic quality, pragmatic quality, capability quality, evolvability quality – and nine metrics based on object-oriented metrics to measure each quality property. Basciani, et al. [11] designed a questionnaire to evaluate the perceived overall quality of the meta-model based on the quality attributes previously defined by [9]. López-Fernández et al. propose 30 common quality criteria for meta-models and an integrated tool for checking them during the meta-model construction [8, 12].

On the other hand, Marín, et al. [13] propose a metamodel for defect detection in Model-Driven Development oriented conceptual models and a set of rules for the detection of defects in the model.

Regarding validation techniques, there is no specific literature focused on validation methods for meta-models. According to the systematic literature review of meta-models for software quality and its evaluation [14], the selected papers applied different validation methods: designing case studies, toy experiments, peer reviews by experts, and a pilot project application. Moreover, the authors highlight the importance of using a real-world case for the validation of any meta-model to demonstrate its usefulness. These methods can be combined, for example, in [15] authors apply empirical validation and involve experts in a peer-reviewed approach. Furthermore, in [16] authors use experts validation joined to a case study.

3. Design of the questionnaire

Although meta-models are prone to extensions and modifications, the main entities and relationships must be identified to ensure the proper development of dashboards, in this case.

The dashboard meta-model has been developed using a domain engineering [17, 18] and an example-driven approach [19], in which different dashboards and data visualizations were analyzed to extract main and standard features among their elements. This methodology allows the identification of abstract qualities that can be arranged into a meta-model to obtain fully functional dashboard instantiations.

We developed the first version of an automatic dashboard generator based on the meta-model to test if it supports the generation of real-world dashboard and visualization examples. The code generator inputs a set of parameters that account for the elements and attributes of the meta-model, and the result is the source code of a dashboard according to the provided configuration.

The approach taken to automatically generate the source code is based on the software product line (SPL) paradigm [20, 21] and we developed different HTML and JavaScript code templates [22] to materialize the variability points of the product line [23].

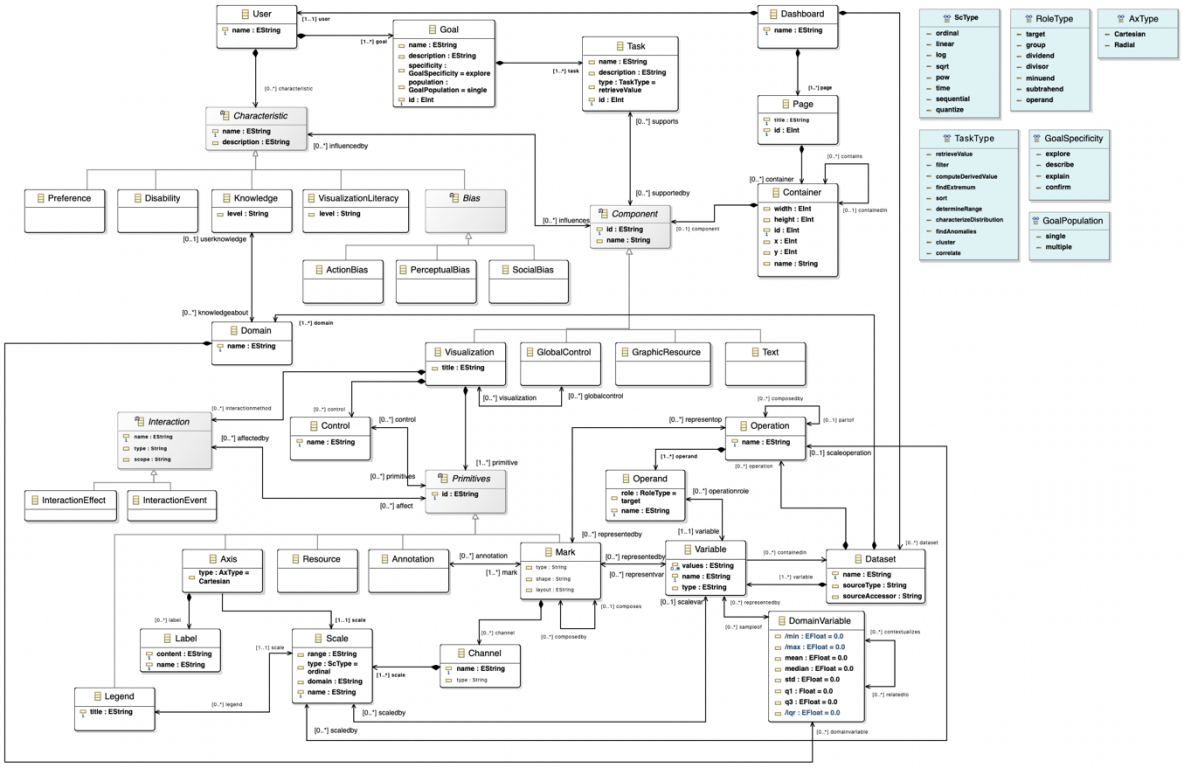

Figure 1: The current version of the dashboard meta-model. A high-resolution version of the image can be consulted at https://zenodo.org/record/7037624.

During the development and testing of the dashboard generator, the meta-model was subject to modifications because this process raised new dimensions and relationships that need to be accounted for during the design and development of these tools. Figure 1 presents the current version of the dashboard meta-model.

We created an online questionnaire in which six different sections of the meta-model (dashboard layout, user characteristics, goals and tasks, user and dashboard relationships, data visualization primitives, and data domain and operations), in addition to the whole meta-model, were scored in terms of the mentioned dimensions using a 1-4 scale, where one implies that the section does not meet the criterion, and four that it highly meets the criterion. Figure 2 shows the rubric used to score the different dimensions based on previous works on content validation by experts [25].

Criterion	Does not meet the criterion	Low	Medium	High
Clarity	The meta-model section is not clear	The meta-model section requires modifications on the relationships or entities	The meta-model section requires specific modifications on some relationships or entities	The meta-model section is clear, and the relationships and entities capture the dashboards' domain characteristics
Coherence	The meta-model section has no logic relationship with the dashboards' domain	The meta-model section has a slight logic relationship with the dashboards' domain	The meta-model section has a moderate logic relationship with the dashboards' domain	The meta-model section is completely related with the dashboards' domain
Relevance	The meta-model section could be removed without affecting the meta-model as a whole	The meta-model section has some relevance, but it is not especially relevant to define dashboards	The meta-model section is relatively important to define dashboards	The meta-model section is crucial to define dashboards

Figure 2: The employed rubric to score the meta-model's content.

We also added a brief explanation of the meta-model section and a "Yes/No" question to test if the representation meets the intended goal of the section. Finally, we included an open text field to collect any qualitative feedback or justification that experts might have. The questionnaire was configured in a customized installation of LimeSurvey and sent by e-mail to different domain experts from both the software engineering and data visualization fields.

4. Discussion and conclusions

This paper presents our work-in-progress for validating a meta-model framed in the data visualization and dashboard domain. The meta-model is set to be used as a foundational learning resource for users to understand the primitive elements of data visualizations, which is, in turn, important to reach proper insights.

The meta-model has been subject to continuous modifications following the domain engineering approach, including not only tangible and visual elements but also abstract concepts related to the user and data domain. The resulting outcome is a complex meta-model (Figure 1) that tries to enclose every primitive element involved in the data visualization domain.

Due to the meta-model's inherent complexity and our intention to use it as a learning resource, we proposed a content validation of this artifact before integrating it into real-world processes.

The validation questionnaire has been designed to measure the coherence, relevance, and clarity of the dashboard meta-model. It was necessary to divide the meta-model into different sections to ease the analysis. This questionnaire has been sent to experts in the data visualization and software engineering domain with the goal of obtaining different perspectives about the content of the meta-model.

The measured dimensions will provide crucial information to improve the meta-model, which will be translated into a better learning resource that provides the most relevant elements in data visualizations in an understandable, clear, and coherent manner.

Future research will involve the analysis of the questionnaire responses as well as the improvements related to the potential issues that might arise from the expert validation.

Once the validation has been carried out, we plan to use the meta-model as a learning resource by implementing an educational tool focused on improving the understandability of the elements involved in data visualizations and information dashboards.

5. Acknowledgements

This research work has been supported by the Spanish Ministry of Education and Vocational Training under an FPU fellowship (FPU17/03276). This research was partially funded by the Spanish Government Ministry of Economy and Competitiveness through the DEFINES project grant number (TIN2016-80172-R) and the Ministry of Science and Innovation through the AVisSA project grant number (PID2020-118345RB-I00).

6. References

[1] A. Álvarez-Arana, M. Villamañe-Gironés, and M. Larrañaga-Olagaray, "Improving Assessment Using Visual Learning Analytics," *Education in the Knowledge Society,* vol. 21, Article 9, 2020, doi: https://doi.org/10.14201/eks.21554.

[2] A. G. Kleppe, J. Warmer, and W. Bast, "MDA Explained. The Model Driven Architecture: Practice and Promise," ed: Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 2003.

[3] A. Vázquez-Ingelmo, A. García-Holgado, F. J. García-Peñalvo, and R. Therón, "A Meta-Model Integration for Supporting Knowledge Discovery in Specific Domains: A Case Study in Healthcare," *Sensors,* vol. 20, no. 15, p. 4072, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/15/4072.

[4] A. Vázquez-Ingelmo and R. Therón, "Beneficios de la aplicación del paradigma de líneas de productos software para generar dashboards en contextos educativos," *RIED. Revista Iberoamericana de Educación a Distancia,* vol. 23, no. 2, pp. 169-185, 07/01 2020, doi: 10.5944/ried.23.2.26389.

[5] A. Vázquez-Ingelmo, F. J. García-Peñalvo, and R. Therón, "Generating Dashboards Using Fine-Grained Components: A Case Study for a PhD Programme," in *Learning and Collaboration Technologies. Designing, Developing and Deploying Learning Experiences*, Cham, P. Zaphiris and A. Ioannou, Eds., 2020// 2020: Springer International Publishing, pp. 303-314, doi: 10.1007/978-3-030-50513-4_23.

[6] A. Vázquez-Ingelmo, F. J. García-Peñalvo, and R. Therón, "Taking advantage of the software product line paradigm to generate customized user interfaces for decision-making processes: a case study on university employability," *PeerJ Computer Science,* vol. 5, e203, 2019, doi: 10.7717/peerj-cs.203.

[7] B. W. Boehm, "Verifying and validating software requirements and design specifications," *IEEE Software,* vol. 1, no. 1, pp. 75-88, 1984.

[8] J. J. López-Fernández, E. Guerra, and J. de Lara, "Combining unit and specification-based testing for meta-model validation and verification," *Information Systems,* vol. 62, pp. 104-135, 2016/12/01/ 2016, doi: https://doi.org/10.1016/j.is.2016.06.008.

[9] M. F. Bertoa and A. Vallecillo, "Quality attributes for software metamodels," presented at the Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE 2010), Málaga, Spain, 2010.

[10] Z. Ma, X. He, and C. Liu, "Assessing the quality of metamodels," *Frontiers of Computer Science,* vol. 7, no. 4, pp. 558-570, 2013/08/01 2013, doi: 10.1007/s11704-013-1151-5.

[11] F. Basciani, J. Di Rocco, D. Di Ruscio, L. Iovino, and A. Pierantonio, "A tool-supported approach for assessing the quality of modeling artifacts," *Journal of Computer Languages,* vol. 51, pp. 173-192, 2019/04/01/ 2019, doi: https://doi.org/10.1016/j.cola.2019.02.003.

[12] J. J. López-Fernández, E. Guerra, and J. de Lara, "Assessing the Quality of Meta-models," in *MoDeVVa*, vol. 1235, F. Boulanger, M. Famelis, and D. Ratiu Eds., no. CEUR Workshop Proceedings Series). Valencia, Spain: CEUR-WS.org, 2014, pp. 3-22.

[13] B. Marín, G. Giachetti, O. Pastor, and A. Abran, "A Quality Model for Conceptual Models of MDD Environments," *Advances in Software Engineering,* vol. 2010, p. 307391, 2010/07/27 2010, doi: 10.1155/2010/307391.

[14] N. Yılmaz and A. K. Tarhan, "Meta-models for Software Quality and Its Evaluation: A Systematic Literature Review," in *Joint Proceedings of the 30th International Workshop on Software Measurement and the 15th International Conference on Software Process and Product Measurement (IWSM Mensura 2020) (Mexico City, Mexico, October 29-30, 2020)*, vol. 2725, A. Abran and Ö. Özcan-Top Eds., no. CEUR Workshop Proceedings Series): CEUR-WS.org, 2020.

[15] S. Wagner *et al.*, "Operationalised product quality models and assessment: The Quamoco approach," *Information and Software Technology,* vol. 62, pp. 101-123, 2015/06/01/ 2015, doi: https://doi.org/10.1016/j.infsof.2015.02.009.
[16] P. Mohagheghi and V. Dehlen, "A metamodel for specifying quality models in model-driven engineering," in *Proceedings of the Nordic Workshop on Model Driven Engineering*, 2008, pp. 51-65.
[17] G. F. Arango, "Domain engineering for software reuse," 1988.
[18] K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Novak, and A. S. Peterson, "Feature-oriented domain analysis (FODA) feasibility study," Carnegie-Mellon University, Software Engineering Institute, Pittsburgh, PA, USA, 1990.
[19] J. J. López-Fernández, J. S. Cuadrado, E. Guerra, and J. De Lara, "Example-driven meta-model development," *Software & Systems Modeling,* vol. 14, no. 4, pp. 1323-1347, 2015.
[20] P. Clements and L. Northrop, *Software product lines*. Addison-Wesley, 2002.
[21] K. Pohl, G. Böckle, and F. J. Van der Linden, *Software Product Line Engineering: Foundations, Principles and Techniques*. New York, NY, USA: Springer-Verlag New York, Inc., 2005.
[22] A. Vázquez-Ingelmo, F. J. García-Peñalvo, and R. Therón, "Addressing Fine-Grained Variability in User-Centered Software Product Lines: A Case Study on Dashboards," in *World Conference on Information Systems and Technologies*, 2019: Springer, pp. 855-864.
[23] A. Vázquez-Ingelmo, F. J. García-Peñalvo, and R. Therón, "Taking advantage of the software product line paradigm to generate customized user interfaces for decision-making processes: a case study on university employability," *PeerJ Computer Science,* vol. 5, p. e203, 2019/07/01 2019, doi: 10.7717/peerj-cs.203.
[24] R. Skjong and B. H. Wentworth, "Expert judgment and risk perception," in *Proceedings of the Eleventh (2001) International Offshore and Polar Engineering Conference (Stavanger, Norway, June 17-22, 2001)*: International Society of Offshore and Polar Engineers, 2001.
[25] J. Escobar-Pérez and A. Martínez, "Validez de contenido y juicio de expertos: Una aproximación a su utilización," *Avances en Medición,* vol. 6, pp. 27-36, 01/01 2008.