Generation of User Interfaces from Business Process Model Notation (BPMN)

Eduardo Diaz, Silvia Rueda
Escola Tècnica Superior d'Enginyeria, Departament d'Informàtica, Universidad de Valencia, Avenida de la Universidad, s/n, 46100, Burjassot
València, Spain
diazsua@alumni.uv.es,
silvia.rueda@uv.es

EICS '19, June 18–21, 2019, Valencia, Spain

ACM ISBN 978-1-4503-6745-5/19/06…$15.00
https://doi.org/10.1145/3319499.3328242

Abstract
Business Process Model Notation focuses on functional processes; so, the design of the interface generally depends on the subjective experience of the analyst. This thesis proposes a new method to generate interfaces from BPMN models. The idea is to identify rules from BPMN to interfaces in existing real projects. We have analyzed 7 Bizagi projects to generalize a list of rules. It has been done considering five BPMN patterns. Apart from BPMN primitives, there are rules that depend on elements of Class Diagrams to know how to generate the interfaces. When the rules have several alternatives to generate the interfaces, we need an unambiguous semantics to specify which alternative we are going to use. We propose extending the BPMN model with new stereotypes to specify when using each alternative. Which alternatives could improve the usability among all the possibilities is also a target of study in the thesis.

Author Keywords
BPMN; user interfaces; patterns; rules; stereotypes.

Introduction
Business Process Model Notation (BPMN) [1] provides organizations with the ability to understand their

internal business procedures in a graphical notation, as well as the ability to communicate these procedures in a standard way. The basic conceptual primitives in BPMN models are events, gateway, swimlane, flows, and tasks [1]. There are two types of tasks: user type task (carried out by a person or user with the help of a system or software) and service type task (carried out by a system without human intervention, for example an automatic task). We used these tasks since they are the most frequently used in BPMN. Currently, interfaces are currently derived from BPMN models in a traditional way depending on the analyst experience. This means that the effort made building the BPMN is not useful to design the interfaces. In addition, normally, analysts that build the BPMN models are not the same designers who implement the user interface, generating a gap between what is described in the BPMN models and what it is really implemented in the interface. In this thesis, we propose a method to generate interfaces from BPMN models. The approach is based on the study of five widely used patterns in business process models. We have studied the use of these five patterns [2] in Bizagi projects [3]; these projects deal with problems of administration sales and, education, among others. Bizagi repository has both the BPMN models and the implementation of the interfaces, so we can analyze the mapping between both elements. Analyzing how each BPMN model results in an interface, we can extract transformation rules that can be generalized for any project. When the rules have several alternatives to generate the interfaces, we need an unambiguous semantic to specify which alternative will be used. For this aim, we have extended the BPMN model with new stereotypes that allow to specify what alternative will be applied in each case. The stereotypes of the BPMN model have been also complemented with stereotypes of UML Class Diagram, since part of the interface depends on persistency model. Apart from the rules, we aim to define a set of recommendations to know what alternatives are more recommended to improve the system usability. The thesis aims to be validated throughout three experiments. (1) Validation of the rules extracted from the Bizagi projects; (2) Validation of the recommendations to improve the usability; (3) Validation of the method to generate interfaces from BPMN models through the development of a tool that implements the model to code rules.

Related work

This section reviews previous works related with our proposal: the user interface generation from BPMN. Marco Brambilla et al. [4] defined a process modeling in web applications. They present new web engineering methods for the high-level specification of applications that present business processes and remote service innovation in the tool WebRatio [5]. Lei Han et al. [6] defined an approach of derivation of user interfaces from BPMN models based on a role-enriched business process model developed with tasks descriptions and associated data. Kenia Sousa et al. [7] defined a model driven approach for organizational engineering in which user interfaces are derived from business processes. The approach is based on the Cameleon Reference Framework [8]. Chun Ouyang et al. [9] defined transformation rules between BPMN models and BPEL definitions. This approach includes an integrated set of techniques to translate the captured models using a central subset of patterns BPMN to generate code Business Process Execution Language for web services. Wided Bouchelligua et al. [10] defined a user interface modeling based on workflows, the approach follows a set of model transformations according to a Model

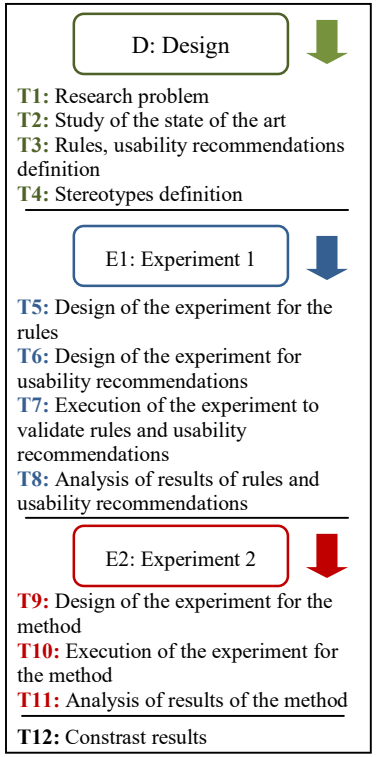

Figure 1: shows the tasks that compose the Design of our approach (D), and the tasks of two experiments, E1 to validate rules and usability recommendations and E2 to validate the code generation of the method.

Driven Engineering approach. Javier Gonzales et al. [11] defined an approach to transform BPMN models to software artefacts. The approach consists in three steps: (1) refining BPMN models through reengineering; (2) automating patterns to derive software analysis; (3) designing artifacts UML.

Problem statement

BPMN models are only focused on functional behavior, the interface generation depends on developers preferences. In order to systematize and automate the process as much as possible, we propose a set of model to code transformation rules to generate interfaces from BPMN conceptual primitives. The approach is complemented with the use of UML Class Diagrams to know what widgets must appear in each interface. We propose extending the BPMN notation with new stereotypes to specify what widget must be generated for each primitive of the BPMN model.

Research questions

We aim to answer three research questions:

RQ1: What generation rules can transform BPMN models into user interfaces? We define a set of rules to generate user interfaces.

RQ2: From all the possible alternatives for each rule, which one is the most suitable to improve the usability of the interfaces? We define usability recommendations for each alternative of the rules.

RQ3: What stereotypes are needed to extend BPMN for generating user interfaces? We define stereotypes with an unambiguous semantics to specify which alternative of the rule we are going to use in the generation.

Research approach

This section deals with the different methodological aspects of the thesis. The type of research of this thesis corresponds to "Design Science", since it aims to design a new artifact, through action and decision based on a set of systematic tests [12]. Design science is a methodology based on the design and investigation of designed to interact with a problematic context in order to improve something in that context [13]. In this thesis, the new artifact is the method to generate user interfaces, and the context is BPMN models. The Design Science consists in two main activities: Design (D) and Experimentation (E). Tasks related with design identify a problem and propose a solution, while tasks related with experimentation design and validate an experiment to validate the solution.

Figure 1 shows, **D: T1** is tackled in section Problem Statement and **T2** is described in section State of the art of this document. **T3**: Taking as input 7 Bizagi projects that include BPMN models and interfaces, we generalize the rules used in them to transform BPMN models into interfaces. In order to perform the process in a systematic way, we propose defining rules per BPMN pattern. We focus our study on 5 well known patterns: Sequence, Exclusive decision, synchronization, implicit decision and structure union synchronization [2]. Note that each rule has several alternatives; the choice of a concrete alternative depends exclusively on the analyst, which may be a subjective decision. In order to help the analyst in the choice to get usable interfaces, we propose classifying the alternatives according to usability recommendations previously defined in the literature. Using our classification, the analyst can choose which alternative is more usable without being an expert at usability. **T4**:

Some rules have several alternatives to generate different widgets, we propose defining one stereotype per alternative. For example, stereotype << U >> generates a Form.

E1: T5: Subjects draw user interfaces from BPMN models and a class diagram according to their preferences. This way we can compare whether the rules applied unconsciously by the subjects match with our proposed rules. **T6**: We evaluate whether our recommended alternatives (defined in T3) improve the usability comparing subjects' criteria with our recommendations. **T7**: The experiment to validate the rules and the usability recommendations is conducted with students of a subject in HCI in the degree of Computer Engineering. **T8**: We analyze results of the experiment in terms of correctness [14] and completeness [14], perceived usefulness and intention to use [15].

E2: T9: We design an experiment to validate the rules defined in T3 for generating interfaces with a tool within Visual Paradigm [16]. A group of subjects generates user interface from a standard BPMN model and other group of subjects generate user interface with our proposal (with stereotypes). At the end we compare both approaches comparing Efficiency, Efficacy and Satisfaction [14]. **T10**: We conduct the experiment designed in T9 with subjects of the subject of Software Engineering in the degree of Computer Engineering. **T11**: We analyze the results of the experiment and propose some recommendations [14].

T12: The results obtained from Experiment 1 and Experiment 2 are analyzed and provide input to improve our approach.

Results to date

Currently we have defined a set of 14 rules. A first version of these rules was published in [17], this first version used three patterns: sequence, exclusive decision and synchronization. Also a second version of the rules was published in [18], this second version used two patterns, implicit decision and synchronization union structure. The experiment E1 has been already conducted with students of the University of Valencia and nowadays we are analyzing the data. We have already implemented a tool based on Visual Paradigm to generate interfaces in PHP from BPMN models. The tool supports the proposed stereotypes (http://hci.dsic.upv.es/bpmn/). This tool does not store data, only generates non-functional interfaces.

Contributions

RQ1 and RQ2 are answered through Experiment 1, while RQ3 is answered through Experiment 2. The main contribution of the thesis can be summarized in: (1) The definition of a method to generate interfaces from BPMN models; (2) A proposal of stereotypes to enrich the BPMN to specify what widget will represent graphically each BPMN primitive; (3) Implementation of a tool based on Visual Paradigm to generate interfaces from BPMN models; (4) Validation of the generation rules; (5) Validation of the alternatives of the rules that are recommended to improve the usability; (6) Validation of the code generation from BPMN models.

Acknowledgements

This thesis has the support of Generalitat Valenciana through IDEO (PROMETEOII/2014/039), Spanish Ministry of Science and Innovation through DataME (ref: TIN2016-80811-P) and Ministry of Education of Peru, PRONABEC President of the Republic Scholarship.

References

[1] Group, O. M. *Business Process Modeling Notation, http://www.bpmn.org*. OMG, 2017.

[2] BPMN, B. *Business model patterns, http://resources.bizagi.com/docs/Workflow_Patterns_using_BizAgi_Process_Modeler_Esp.pdf*. 2017.

[3] Bizagi *Examples of BPMN Projects, https://www.bizagi.com/es/comunidad/process-xchange*. 2017.

[4] Brambilla, M., Ceri, S., Fraternali, P. and Manolescu, I. Process modeling in Web applications. *ACM Trans. Softw. Eng. Methodol.*, 15, 4 2006), 360-409.

[5] Granada, D., Vara, J. M., Brambilla, M., Bollati, V. and Marcos, E. Analysing the cognitive effectiveness of the WebML visual notation. *Software and Systems Modeling*, 16, 1 2017), 195-227.

[6] Han, L., Zhao, W. and Yang, J. An approach towards user interface derivation from business process model. *Communications in Computer and Information Science*, 6022016), 19-28.

[7] Sousa, K., Mendonça, H. and Vanderdonckt, J. *User Interface Derivation from Business Processes: A Model-Driven Approach for Organizational Engineering*. Springer, 2007.

[8] Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L. and Vanderdonckt, J. A Unifying Reference Framework for multi-target user interfaces. *Interacting with Computers*, 15, 3 2003), 289-308.

[9] Ouyang, C., Dumas, M., Aalst, W. M. P. V. D., Hofstede, A. H. M. T. and Mendling, J. From business process models to process-oriented software systems. *ACM Trans. Softw. Eng. Methodol.*, 19, 1 2009), 1-37.

[10] Bouchelligua, W., Mahfoudhi, A., Mezhoudi, N., Daassi, O. and Abed, M. *User interfaces modelling of workflow information systems*. 2010.

[11] Gonzalez-Huerta, J., Boubaker, A. and Mili, H. A business process re-engineering approach to transform BPMN models to software artifacts. *Lecture Notes in Business Information Processing*, 2892017), 170-184.

[12] Wieringa, R. J. *Design science methodology for information systems and software engineering*. Springer, 2014.

[13] Wieringa, R. Empirical research methods for technology validation: Scaling up to practice. *Journal of Systems and Software*, 95, 0 2014), 19-31.

[14] IEEE *Systems and software engineering -- Vocabulary*. 2010.

[15] Moody, D. L. The method evaluation model: a theoretical model for validating information systems design methods. In *Proceedings of the European Conference on Information Systems (ECIS 03)* (Naples, Italy 2003).

[16] Paradigm, V. *Visual Paradigm, https://www.visual-paradigm.com/*. 2004.

[17] Diaz, E., Panach, J. I., Rueda, S. and Pastor, O. Towards a method to generate GUI prototypes from BPMN. In *Proceedings of the 2018 12th International Conference on Research Challenges in Information Science (RCIS)* (29-31 May 2018, 2018). IEEE.

[18] Diaz, E., Panach, J. I., Rueda, S. and Pastor, O. Generación de Interfaces de Usuario a partir de Modelos BPMN con Estereotipos. In *Proceedings of the Jornada de la Sociedad de Ingeniería de Software y Tecnologías de Desarrollo de Software (SISTEDES)* (17 - 19 Setiembre 2018, 2018). Sistedes.