HAL
open science

Toward Requirements-Driven Design of Visual Modeling Languages

Jens Gulden, Eric Yu

To cite this version:

Jens Gulden, Eric Yu. Toward Requirements-Driven Design of Visual Modeling Languages. 11th IFIP Working Conference on The Practice of Enterprise Modeling (PoEM), Oct 2018, Vienna, Austria. pp.21-36, 10.1007/978-3-030-02302-7_2 . hal-02156474

HAL Id: hal-02156474

https://inria.hal.science/hal-02156474v1

Submitted on 14 Jun 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Toward Requirements-Driven Design of Visual Modeling Languages

Jens Gulden[1] and Eric Yu[2,3]

[1] University of Duisburg-Essen, Essen, Germany
[2] Faculty of Information, University of Toronto, Toronto, Canada
[3] Department of Computer Science, University of Toronto, Toronto, Canada
jensgulden@acm.org
eric.yu@utoronto.ca

Abstract. The design of a visual modeling language demands for a large number of decisions to be taken, depending on the intended purposes of the language, the domain context, and the goals and requirements of different stakeholders who are the prospective users of the language. Methodical support for the design and choice of visual modeling languages plays an important role in Enterprise Modeling (EM), because EM strongly relies on the use of visual modeling languages for expressing human-understandable abstractions of complex domain contexts.
However, existing research primarily discusses individual design aspects of visual modeling languages. The results of these studies partially overlap or contradict each other. The work at hand introduces an approach for systematically identifying and managing trade-offs between competing design recommendations, as well as for gaining an integrated multi-perspective view on requirements towards visual modeling languages.
We demonstrate the feasibility of the approach by reconsidering some design decisions taken for the widely used Business Process Modeling and Notation (BPMN) language.

Keywords: Visual Notation · Visual Modeling Languages · Goal Modeling · Requirements · Soft-Goals

1 Requirements-driven Design to Guide Design Decisions for Visual Languages

With the increasing use of modeling techniques in science and practice of Enterprise Modeling and other fields, the demand for advanced visual modeling languages that support modeling tasks more easily and efficiently is increasingly discussed in the literature [13,16,17].

A large amount of research from an Information Systems perspective has been performed already on the use and design of visual modeling languages, especially on process modeling languages [21,4,8]. This has led to a respectable amount of research questions that have been addressed in the body of literature about individual aspects of visual modeling languages and prescriptive principles suggested for the design of visual modeling languages [2,22,24,9].

While the existing body of research addresses a wide range of isolated design questions, dealing with a set of individual principles does not provide sufficient support for guiding design decisions. This is because in order to develop a visual language as a whole, diverse aspects about the purpose of the language, the prospective users of the language, and their intentions, have to be brought into a coherent balance. Individual recommendations for addressing design principles may contradict each other. The issues they address may also overlap, or there may be blank spots which are not addressed by research yet. In order to apply existing research on visual modeling language design for creating entirely new visual languages, or for extending existing ones in a justified and systematic way, it is thus necessary to have methodical support for discovering contradictions, overlaps, and ambiguities among existing individual design principles, and to have guidance in resolving them.

Especially the fact that many modeling languages are intended to serve as interfaces between different stakeholder groups, e. g., business experts, software developers, or novices to be trained in any of these fields, naturally leads to the situation that in order to fulfill each of these groups' information needs, trade-offs will arise when deciding for the design of a language. Identifying these trade-offs, and resolving conflicts resulting from them, are important tasks in a reflected visual language design process. However, up to now elaborate methodical support for identifying and resolving trade-offs among individual design principles for visual languages, has not been proposed in the modeling literature. As a consequence, the research question addressed in this article is:
How can trade-offs among individual design principles for visual modeling languages be systematically elicited and resolved, to guide justified design decisions during the process of creating, extending, or evaluating visual modeling languages?

In order to provide an answer to this question, we present an approach that treats properties of visual languages and individual design principles as design goals that can systematically be analyzed and made traceable to guide the design choices for a visual language.

The remainder of the article is structured as follows: In Sect. 2 the demand for an integrating perspective on visual modeling language design and systematic support for deriving design decisions from requirements in a justifiable and traceable manner is laid out. Related work is discussed in Sect. 3. In Sect. 4 the proposed approach is elaborated and exemplified on top of the scientific knowledge body of research on visual process modeling languages. The applicability of the approach is evaluated in Sect. 4.4 by using it to reconsider design decisions that had been taken for the BPMN 2.0 language. Sect. 5 completes the presented considerations with a conclusion and an outlook on future work.

2 Demand for Goal-guided Design of Visual Modeling Languages

2.1 Principle-driven design of visual languages is not enough

The design of a visual modeling language is a complex process which demands designers to find a balance between diverse criteria that a language should adhere to. Among them are the intended capabilities of the language to express concepts of a particular domain, if a domain-specific language is to be developed. Secondly, modeling languages are in most cases used as communication tools among different groups of stakeholders who are the prospective users of the language. They need to be able to apply the modeling language in accordance with their purposes, cognitive skills, and experiences, which may differ strongly depending on the involved types of stakeholders. It may also turn out that a single language is not sufficient for all involved stakeholders, and that multiple visual languages are required which reflect the same underlying semantics. As a third aspect, additional factors such as integration capabilities into a set of other existing languages, or demands for automatic analysis and further processing of models, are likely to play a role and to have influence on the design decisions. As a consequence, the question which is the "right" visual modeling language, and in turn the diverse design questions that come up during the process of specifying a visual modeling language, cannot be answered in the same way for every language.

Existing research has provided a wide variety of detail examinations on individual design principles that play a role for designing visual languages (see Sect. 3). As it is paradigmatically normal for scientific work, every single contribution provides detailed examinations of clearly marked individual research questions. This leads to the situation that most available research is about individual aspects of visual language design, and only a single or few design principles are discussed at a time by each examination.

However, given the strong influence of multiple external criteria such as domain concepts, stakeholder purposes, and language infrastructure demands, a visual language design project has to take into account a high amount of different design aspects. As a consequence, contradictions between proposed design principles will occur, and it will not be possible to follow all existing design principles simultaneously.

For example, when faced with the decision of what symbols to choose for representing concepts in a visual domain-specific language, it may be of help to restrict the symbol design to clearly distinguishable abstract geometrical shapes, in order to provide unique symbols that are easy to recognize for experienced language users. Such a decision would support the design goals of semiotic clarity and perceptual discriminability discussed in [22]. On the other hand, picture-like figurative symbols are likely to be more easily understandable by untrained users, and depending on the domain context, could contribute to a higher level of identification of the users with the visual language and lead to an increased acceptance of the language. Choosing this option would support the design goal

of semantic transparency according to [22]. Thus, there is a trade-off between using simple geometrical shapes and speaking pictures as visual symbols, which results from competing and sometimes even contradictory design goals for visual languages.

A method that supports justified visual language design should allow to handle trade-offs of this kind in a way that the choice for a design option can be rationally traced along the criteria that determine a specific language design decision. Especially, diverting demands of heterogeneous stakeholder groups require to resolve multi-criterial design decisions by finding an appropriate balance of trade-offs. Faced with these challenges, the approach presented in this article does not operate on the level of individual design principles, but provides means to systematize multiple design principles in a model-based way, and gives guidance in performing trade-offs and resolving contradictions among them. As a methodical tool for this purpose, the Non-Functional-Requirements (NFR) Framework is employed.

2.2 The NFR Framework

The Non-Functional-Requirements (NFR) Framework provides a methodical approach for reasoning about requirements in terms of goals, decomposed sub-goals, and tasks that are concrete means to fulfill goals [3,5,6]. In the NFR Framework, these elements are represented in a model-based way. Using a lean visual language, requirements can be expressed as ovals that represent goals, while decomposition relationships between the goals are shown as connecting arrows. Fig. 1 shows an example NFR model in the notation used throughout this article, with exemplary goals and tasks from the car design domain.

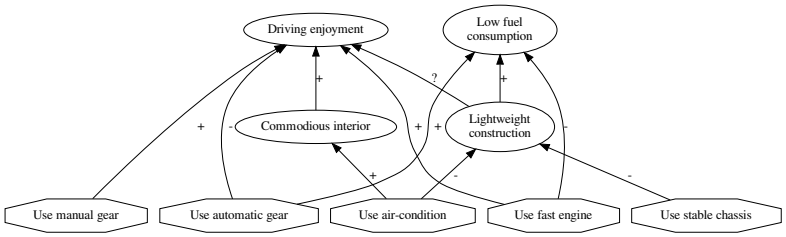

Fig. 1: Example NFR model with language elements used in this article

Relationships between goals can be supportive, in which case the fulfillment of one goal is expected to lead to the fulfillment of another goal, or obstructive, which means fulfilling one goal stands in contrast to fulfilling another goal. Supportive relationships are marked with a '+', obstructive ones with a '-'. Additionally, the relationship between two goals can also be marked as unknown using a '?'.

The bottom row in Fig. 1 contains task elements in octagon shape, which represent possible options to fulfill goals. A decomposition relationship between

a goal and associated tasks indicates that if a task is executed, it contributes to the fulfillment of the goal.

All relationships in an NFR model can additionally be attached with claims, which are comments on the decision rationale that has led to the choice for including a particular relationship in the model. Claims provide support in tracing back the chain of justification for design decisions taken, when an NFR model is applied to meet a set of given requirements for the design of a visual modeling language. When included in a diagram, claims are referenced by numbers in round braces that are placed below the '+/-/?' markers of a relationship line.

Fig. 2 (a) shows a legend of the visual notation for NFR models used in this article. A meta-model that formally describes the abstract syntax of this modeling language is displayed in Fig. 2 (b).

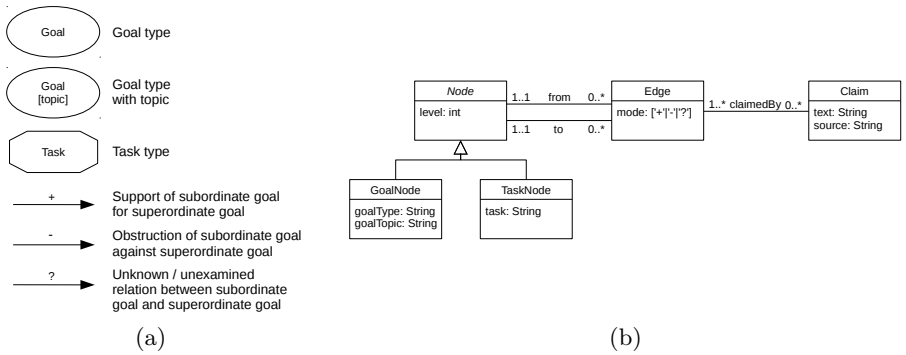

Fig. 2: Legend (a) and meta-model (b) of the NFR modeling language used in this article

Creating NFR models involves performing hierarchical refinements from top-level goals to sub-goals, and to identify task candidates as recommendations for how to achieve the respective sub-goals. This way, a justified chain of reasoning gets documented in a complex structure, which would not be possible purely on a textual basis without a dedicated modeling technique. Furthermore, creators and users of NFR models are continuously motivated to question to what extent an NFR model is coherent and complete, which motivates to find a "good" system of goals.

The refinement of goals along the hierarchy in NFR models can be performed in two conceptual directions: One way is to more specifically focus on aspects that constitute a goal, i. e., a superordinate goal such as "Understandability" gets refined into subordinate goals that each are distinct aspects of it, e. g., "Speed of comprehension", "Accuracy of understanding", and "Learnability". The decomposition of a goal in such a way is called *type refinement*. Another way to decompose goals is to distinguish between different objects of interest to which the same goal is applied. E. g., "Understandability" of a modeling language can be demanded for expert users or novice users, or specific syntactic elements of visual modeling languages can be further considered with respect to their

understandability, e. g., node-symbols, edges, or layout rules of visual languages. A goal refinement which differentiates between multiple objects of interest to which the same goal is applied is called *topic refinement*. It is indicated by adding the topic name in square braces "[ ]" to the label of the refined sub-goal.

3 Related Work

A wide range of scientific literature has contributed to research on the design of visual modeling languages, particularly by examining characteristics that influence the cognitive handling of visual models [2,22,9]. From this work, a variety of design principles and recommendations for designing visual modeling languages have evolved and been empirically addressed. Especially the principles discussed in [22] have shown a large impact on research activities in the scientific community, and have motivated a variety of subsequent empirical works on the cognitive aspects related to these design principles for visual languages [24,26,30]. Our approach makes use of this profound body of research and suggests to take in an integrative perspective, which allows to identify possible contradictions, overlaps, and unexamined aspects among the existing research contributions and in a systematic way.

The basic idea of incorporating a requirements perspective into visual language design has been introduced by a few research publications. [7] suggests an approach for tailoring a requirements engineering approach for the design of visual languages based on the Goal-oriented Requirements Language GRL [1]. The underlying idea of starting visual language design from a goal-oriented perspective is well motivated, however, the elaboration does not incorporate existing research about characteristics of visual languages to support design decisions.

An empiric survey about requirements towards visual language notations expressed by users is conducted by [28]. The examination is oriented along the design principles of the "Physics of Notation" [22], and is based on interviews of modeling experts about their evaluation of the relevance of each principle. But only a few significant statements from the evaluated answers can be generalized, which may be caused by the fact that [22]'s design principles are not easy to be differentiated from the point of view of a modeling language user. For the purpose of applying existing research to language design tasks, [26] proposes a procedure for the systematic application of [22], which is explicated in the form of a process model for principle-based design of a visual modeling language. This operationalization approach is focused on putting design principles into practice, without explicating language users' requirements and corresponding goals.

[29] reflects on the "inherent difficulty of operationalizing the Physics of Notations" [22]. Continuing this work, [30] proposes a framework for verifying visual notation design as a complementary task to developing a language design method based on existing design principles.

In this article, we refer to the existing body of literature to propose an approach which allows to systematically justify the design decisions that are required to be taken when creating or extending a visual modeling language.

4 Applying Requirements-Driven Design Analysis to BPMN

4.1 An illustrative example

Our approach can be used for the design of new visual modeling languages, as well as for creating language extensions and performing justified choices among existing languages to use. To provide a compact demonstration of the approach, we here perform a reconsideration of selected design decisions that have been taken for the Business Process Modeling and Notation (BPMN) 2.0 modeling language [18]. BPMN's visual notation is well established and described in normative documents by the standards organization Object Management Group (OMG). However, the language has grown over time and was compiled from other earlier process modeling languages, without a coherent justification of the appropriateness of each included visual modeling element [14]. This can be demonstrated along an example model issued by the OMG for training and documentation purposes [15] which is shown in Fig. 3. Three areas are marked that reveal examples of possible design deficiencies in the BPMN visual notation, which in larger models have the potential to compromise the comprehensibility and usability of the BPMN language.

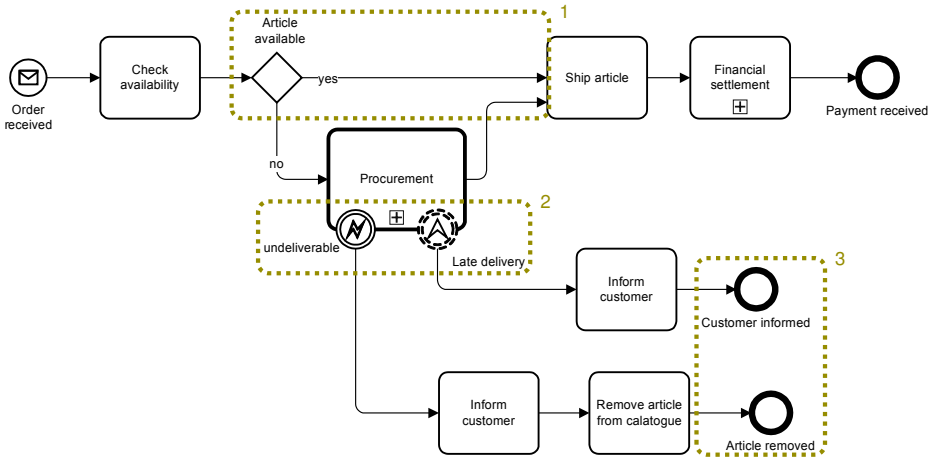

Fig. 3: Order Fulfillment and Procurement example [15] with highlighted problem areas

The area marked with (1) shows a situation where a diverging gateway is used to express the beginning of alternative process flows, but no corresponding converging gateway exists. Instead, the converging merge of the alternative process branches is expressed implicitly by two incoming arrows into the activity "Ship article". While this is syntactically valid in BPMN, there is evidence in the existing body of research that incorporating different notation options for process flows, and especially the combination of a gateway-based and an implicit flow notation, reduces the ease of understanding of BPMN models [24,20].

In the area marked with (2), two example events "undeliverable" and "late delivery" are shown that can occur during the execution of the "Procurement" task. The visual appearance of the "late delivery" event is composed of the symbol for an escalation event with the shape of an upward arrow head, and a double dashed circular border around the symbol which classifies the event as a non-interrupting event. This means the execution of the task does not stop when the event is thrown. The symbol shape of the upward arrow head is close to the shape of the flash symbol that is used for the error event "undeliverable", and the double dashed border is one out of several possible border styles for events, which can be single solid line, double solid line, single dashed line, double dashed line, or thick solid line. There is evidence in research that the choice of this visual appearance of the example events is inappropriate, as both the symbolic content and the border styles, are neither well distinguishable, nor good to memorize [11,28].

The area marked with (3) exemplifies that BPMN does not contain strict guidelines for its secondary notation, which is about the choice of how to place elements onto the diagram plane. The two end-events "Customer informed" and "Article removed" are not vertically aligned, which, although not intended, may inadvertently convey additional semantics about the importance or expected likelihood of the occurrence of these events.

4.2 A goal model on visual language design aspects

To demonstrate our approach, we now construct a hierarchy of goals which on the higher levels represent general requirement towards visual modeling languages, and on the lower levels subsequently get refined to sub-goals and tasks that address the issues in the example case. We will use *type* refinements to decompose higher-level goals into more concrete sub-goals, and *topic* refinements to differentiate between different objects of interest to which a goal can be applied (see Sect. 2.2). Each choice for including tasks in the model will be justified by one or more citations from the body of scientific literature, which are attached as claims to the modeled refinement relationships between goals and tasks.

The goal model is then used to identify design decisions taken for the BPMN 2.0 language that potentially have led to the issues identified in Sect. 4.1. In a final step, the model will be applied to derive possible alternative design solutions with respect to the identified issues. We choose BPMN 2.0 as a language that is well known, in order to not mix the demonstration with language details that would distract from the contribution of the goal-based approach. The in-depth analysis of an existing language with our approach widely resembles the considerations that have to be made when creating entirely new visual languages or create extensions to existing languages, which is why the demonstration covers the full methodical range of our approach.

With respect to process modeling languages and in particular BPMN, a large body of research is available, parts of which will in the following be structured according to the goal-decomposition approach of the NFR framework. As claims

for justification of the constructed goal models, citations from existing principle-based research will be incorporated in the models. We elaborate on the two top-goals "Comprehension" and "Usability" which are are widely addressed in the existing body of research [8].

As a starting point, we decompose the top-goals "Comprehension" and "Usability" into sub-goals which allow to differentiate between different aspects of the general top-goals. These initial decompositions are shown in Fig. 4.

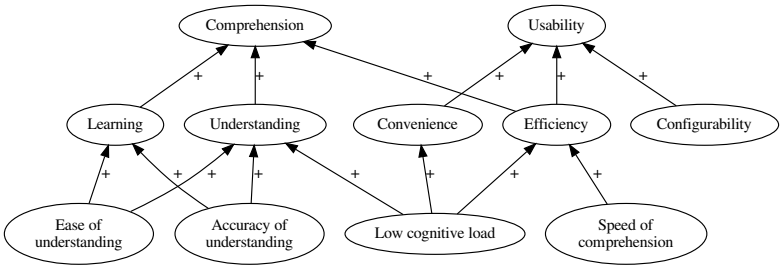

Fig. 4: Goal decompositions that are derived from the top-goals Comprehension and Usability of a visual modeling language

4.3 Comprehension of BPMN

The examination can now drill down to particular sub-goals, in order to relate individual design principles from the body of literature to them. The most common goal investigated by existing research is the effect of visual process modeling languages on the "Comprehension" of models [8]. As a consequence, we focus on refinements of this top-goal in the further examination. Some authors also talk about "cognitive effectiveness" [10,29], which in the context of this article is treated synonymously with comprehensibility. The decomposition of goals that are subsumed under the top-goal of supporting "Comprehension" is shown in Fig. 5.

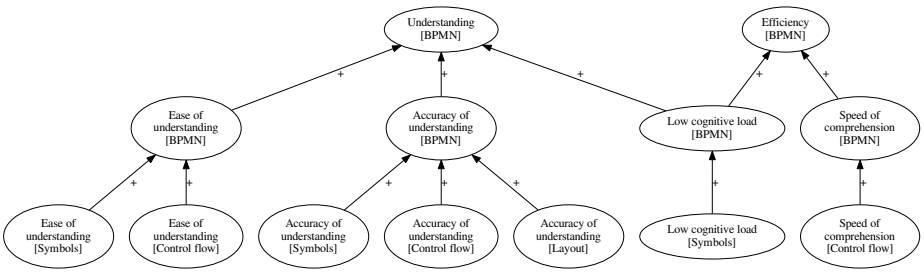

Fig. 5: Decomposition of the top-goal Comprehension

Ease of Understanding We will further focus on the decomposition of "Ease of understanding" as one representative sub-goal of "Understanding", and assign to each detail decomposition relationship at least one claim from the scientific literature on visual process model comprehension. The result is a set of concrete design decisions that can be taken when designing visual modeling languages. These are modeled as task elements in the goal model, indicating that they represent means to fulfill the superordinate goals.

The citations attached to the relationships between goals and tasks are claims that justify the decision to include the relationship in the model. They cite the referenced sources either directly, or are quoted from [8] as a secondary source.

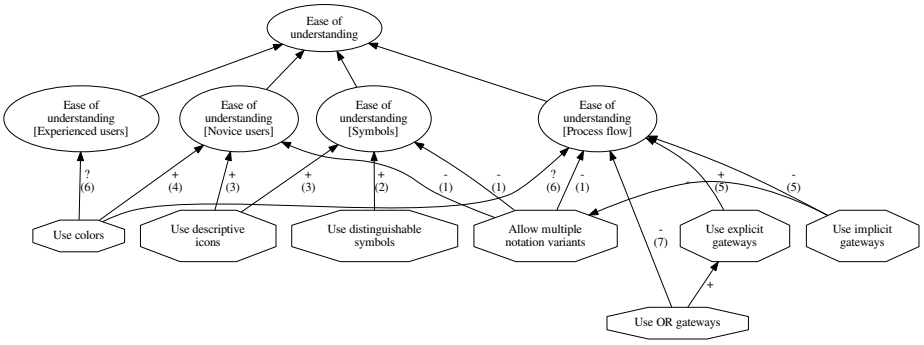

Fig. 6: Decomposition of the goal to ensure ease of understanding into tasks derived from literature references

Figure 6 shows the decomposition hierarchy derived from literature for achieving "Ease of understanding" for visual modeling languages, incorporating a distinction between the topics "BPMN", "Experienced users", "Novice users", "Symbols", and "Process flow". The claims associated with the relationships among the goals are based on the following rationales:

1. *"numerous responses were found that corroborate [the] tendency to require simplicity when dealing with modeling non-experts"* [27]
2. *"the most important requirement [...] is perceptual discriminability"* [28]
3. *"semantic transparency and cognitive fit [...] are indeed vital to ensure non-experts can better understand a visual notation"* [28]
4. *"syntax highlighting with colors for matching gateway pairs is positively related to novices' model comprehension"* [25]
5. *"the use of gateway constructs benefits understanding; implicit representation is often misunderstood"* [24,20]
6. *"no evidence that syntax highlighting improves experts' performance"* [25]
7. *"avoid OR gateway elements"* [21]

Accuracy of Understanding Figure 7 shows the decomposition hierarchy derived from literature for achieving "Accuracy of understanding" for visual modeling languages. The referenced claims are:

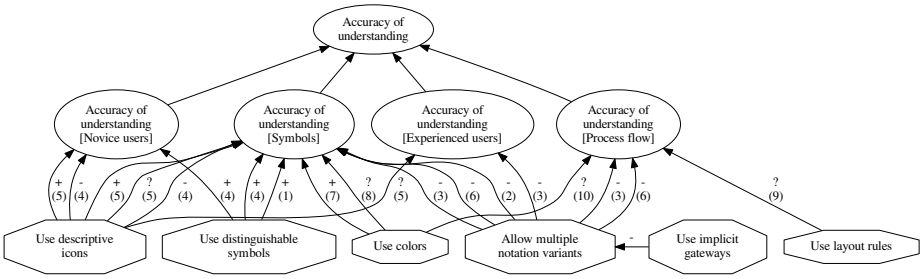

Fig. 7: Decomposition of goals to achieve accuracy of understanding derived from literature references

1. *"perceptual discriminability deficiencies of symbols in YAWL and demonstrate that these symbols lowered comprehension accuracy"* [11]
2. *"semiotic clarity deficiencies in EPC reduce comprehension accuracy"* [11]
3. *"for modeling with experts semiotic clarity [is] perceived as most important"* [28]
4. *"for modeling with non-experts, perceptual discriminability [...] are perceived as most important"* [28]
5. *"semantic transparency and cognitive fit [...] are indeed vital to ensure non-experts can better understand a visual notation"* [28]
6. *"semiotic clarity deficiencies [...] reduce comprehension accuracy"* [11]
7. *"perceptual pop-out and discriminability show their relevance for comprehension accuracy and perceived cognitive load"* [9,11]
8. *"color had no significant main effect on comprehension accuracy."* [19]
9. *"no evidence of the hypothesized superiority of left-to-right flow direction; model elements [...] were made larger and were repositioned [with] no evidence of an effect on comprehension accuracy"* [12,23]
10. *"colored relevant model elements did not significantly affect comprehension accuracy but lowered time taken."* [23]

4.4 Deriving design decisions from the goal models

The approach sketched in the previous sections can be applied to systematically consider design decisions of visual modeling languages, in this case the BPMN. Revising the example given in Fig. 3, the three possible design issues that have been identified can now be addressed using the constructed goal model.

Issue (1) in Fig. 3 was identified because of an ambiguous expression of diverging and converging process flow, the first expressed using the visual element of a gateway, the second implicitly by in-going process flow arrows into a following task element. With respect to the goal "Ease of understanding [Process flow]" in the refinement hierarchy shown in Fig. 6, there are claims that suggest favoring the task "Use explicit gateways" over implicit gateways [24,20]. Furthermore, in the decomposition hierarchy of the "Accuracy of understanding" goal in Fig. 7, the task "Allow multiple notation variants" is claimed to have negative influence on accuracy of understanding [28,11,23], which supports the decision to suggest only one notation variant for process flow divergence and convergence. A possible design alternative that is in accordance with these design principles is shown in Fig. 8 (b).

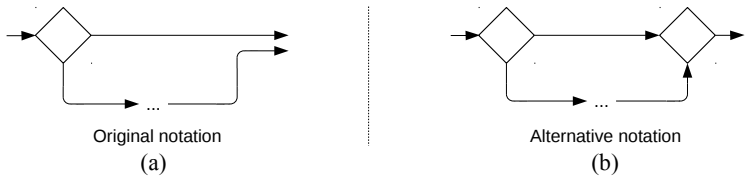

Fig. 8: Original notation (a) and design alternative (b) for issue (1), in accordance with the design principles derived from the goal analysis

With respect to the notation of events, which is addressed by issue (2) marked in Fig. 3, the goal-based analysis reveals that a possible design alternative should consider the tasks "Use distinguishable symbols" and "Use descriptive icons" in the detail models that decompose the "Ease of understanding" and "Accuracy of understanding" goals, since they relate both to the goals "Ease of understanding [Symbols]" and "Accuracy of understanding [Symbols]". These tasks, however, cannot simultaneously be realized, since there is a trade-off between the use of abstract shapes, which are non-descriptive but typically better distinguishable, and the use of picture-like descriptive icons. To resolve this conflict, the relationships of the tasks to their superordinate goals can now be accounted for: In the "Ease of understanding" goal model, both of the tasks "Use distinguishable symbols" and "Use descriptive icons" are almost equally justified by positive claims with respect to the fulfillment of the superordinate goals, with a tendency to "Use descriptive icons" when the goal is to support novice users in achieving "Ease of understanding". The detail model on "Accuracy of understanding", however, unveils that "Use descriptive icons" is only in parts positively associated with its superordinate goals and there are two positive and two negative claims that recommend, respectively advise against, deciding for this task. The "Use distinguishable symbols" task, however, is related to its superordinate goals by supporting relationships only. Given this constellation, a design alternative for the symbols of the error event and the escalation event would take into account to use more simply structured and more clearly distinguishable symbols that are composed of abstract shapes.

An alternative suggestion for a design of the event symbols is shown in Fig. 9 (b). This notation provides a higher level of perceptual discriminability, since the "×" and "!" shapes can be better distinguished by the human cognitive apparatus for visual perception, and they still keep up some level of descriptive meaning, because both symbols are more commonly used metaphors for errors and exceptional situations than the symbols from the original notation. The fact that the process flow continues after the escalation event has been thrown, is accounted for by continuing the process flow notation inside the task element instead of operating with different border styles.

Reasoning about issue (3) in Fig. 3 with the help of the goal-oriented approach allows the conclusion that with respect to understanding the model, a lack of vertical alignment of the end-event symbols does not provide significant disadvantages. This is explicated in the goal model in Fig. 7 by the unspecified

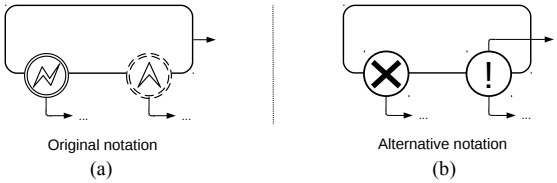

Fig. 9: Original notation (a) and design alternative (b) for issue (2) in accordance with the design principles derived from the goal analysis

'?'-relationship of the goal "Use layout rules" to its superordinate goal "Accuracy of understanding" [12].

5 Conclusion

With the presented approach, we have demonstrated the applicability of the NFR analysis method to the domain of visual modeling language design, along the example of reconsidering a selection of BPMN 2.0 language design decisions. The demonstrated method adopts a model-based approach for a requirements-driven design, in which decisions and corresponding justifications are explicated with a visual formalism and made traceable through the use of corresponding modeling language constructs. This way, trade-offs between existing design alternatives, and multi-perspective design decisions for different groups of stakeholders and language application scenarios can systematically be taken into consideration. As a model-based approach, it is possible to underpin the suggested method with tooling support that implements the proposed approach as software for guiding modeling language developers.

The approach shows how the NFR analysis method serves as a unifying framework to integrate individual parts of existing research. This makes it possible to form a coherent whole out of individual design principles that formerly have been discussed in an isolated way, and provide model-based means to identify and cope with contradictions or incompleteness among the cited literature. It also allows to justify design decisions using explicit, traceable design rationales in the form of claims.

A number of limitations apply to the current state of the elaboration. At first, for now we have concentrated on design decisions regarding the BPMN language only. However, the analysis of an existing language design with our approach widely resembles the considerations that have to be made when creating new visual languages or providing extensions to existing languages. The results of the demonstration thus can be transferred to the tasks of creating or extending any visual modeling language. Furthermore, the visual notation of NFR models as it is used for now leads to a high degree of fragmentation, due to the high amount of hierarchical branches which both can represent type and topic refinements. More research is required on this, and tooling support with interactive features to navigate between multiple perspectives and levels could provide a solution to cope with the complexity of NFR models.

Future work will extend the presented approach by demonstrating it along a wider range of modeling languages, and further elaborate it to a full-fledged framework that is applicable as prescriptive guidance for the creation of new visual modeling languages.

References

1. User requirements notation (URN) language definition. Recommendation Z.151 (11/08). http://www.itu.int/rec/T-REC-Z.151/en, 2008. Accessed on 2018-03-30.
2. J. Aranda, N. A. Ernst, J. Horkoff, and S. M. Easterbrook. A framework for empirical evaluation of model comprehensibility. In *International Workshop on Modeling in Software Engineering at ICSE*, Minneapolis, USA, 5 2007.
3. B. Boehm and H. In. Identifying quality-requirement conflicts. *IEEE Software*, 13(2), March 1996.
4. Andrea Burattin, M. Kaiser, Manuel Neurauter, and B. Weber. Eye tracking meets the process of process modeling: A visual analytic approach. In *Business Process Management Workshops - BPM 2016 International Workshops, Rio de Janeiro, Brazil, September 19, 2016, Revised Papers*, 2016.
5. L. Chung, B. Nixon, and E. Yu. Using non-functional requirements to systematically select among alternatives in architectural design. In *Proc. 1st Int. Workshop on Architectures for Software Systems*, 1994.
6. L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos. *Non-Functional Requirements in Software Engineering*. Springer, Boston Dordrecht London, October 2000.
7. S. de Kinderen and Q. Ma. Requirements engineering for the design of conceptual modeling languages. *Applied Ontology*, 10(1), 2015.
8. K. Figl. Comprehension of procedural visual business process models. *Business & Information Systems Engineering*, 59(1), Feb 2017.
9. K. Figl, J. Mendling, and M. Strembeck. The influence of notational deficiencies on process model comprehension. *Journal of the Association for Information Systems*, 14(6), June 2013.
10. K. Figl, J. Mendling, M. Strembeck, and J. Recker. On the cognitive effectiveness of routing symbols in process modeling languages. In W. Abramowicz and R. Tolksdorf, editors, *Business Information Systems*, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
11. K. Figl, J. Recker, and J. Mendling. A study on the effects of routing symbol design on process model comprehension. *Decision Support Systems*, 54(2), 2013.
12. K. Figl and Mark Strembeck. Findings from an experiment on flow direction of business process models. In *Proceedings of the 6th International Workshop on Enterprise Modelling and Information Systems Architectures (EMISA)*, volume 248 of *Lecture Notes in Informatics (LNI)*. GI, 2015.
13. U. Frank. Domain-specific modeling languages: Requirements analysis and design guidelines, 05 2013.
14. Nicolas Genon, Patrick Heymans, and Daniel Amyot. Analysing the cognitive effectiveness of the BPMN 2.0 visual notation. In *Proceedings of the Third International Conference on Software Language Engineering*, SLE'10, pages 377–396, Berlin, Heidelberg, 2011. Springer.
15. Object Management Group. Bpmn 2.0 by example, version 1.0 (non-normative). https://www.omg.org/spec/BPMN/2.0/About-BPMN/, 6 2010. Accessed on 2018-03-30.

16. J. Gulden and H. A. Reijers. Toward advanced visualization techniques for conceptual modeling. In J. Grabis and K. Sandkuhl, editors, *Proceedings of the CAiSE Forum 2015 Stockholm, Sweden, June 8-12, 2015*, CEUR Workshop Proceedings. CEUR, 2015.
17. J. Gulden, D. v. d. Linden, and B. Aysolmaz. Requirements for research on visualizations in information systems engineering. In *Proceedings of the ENASE Conference 2016, Rome*, 2016.
18. Business Process Management Initiative. Business process modeling notation 2.0 (BPMN 2.0), 2011.
19. T.-F. Kummer, J. Recker, and J. Mendling. Enhancing understandability of process models through cultural-dependent color adjustments. *Decis. Support Syst.*, 87(C), July 2016.
20. H. Leopold, J. Mendling, and O. Gunther. Learning from quality issues of BPMN models from industry. *IEEE Software*, 33(4), July-Aug. 2016.
21. J. Mendling, H. A. Reijers, and W. M. P. van der Aalst. Seven process modeling guidelines (7PMG). *Inf. Softw. Technol.*, 52(2), February 2010.
22. D. L. Moody. The "physics" of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering*, 35(6), 11/12 2009.
23. R. Petrusel, J. Mendling, and H. A. Reijers. Task-specific visual cues for improving process model understanding. *Information and Software Technology*, 79, 2016.
24. J. Recker. Empirical investigation of the usefulness of gateway constructs in process models. *European Journal of Information Systems*, 22(6), 2013.
25. H. A. Reijers, T. Freytag, J. Mendling, and A. Eckleder. Syntax highlighting in business process models. *Decis. Support Syst.*, 51(3), June 2011.
26. M. das Graças da Silva Teixeira, G. K. Quirino, F. Gailly, R. de Almeida Falbo, G. Guizzardi, and M. Perini Barcellos. PoN-S : a systematic approach for applying the Physics of Notation (PoN). In R. Schmidt, W. Guédria, I. Bider, and S. Guerreiro, editors, *Lecture Notes in Business Information Processing*, volume 248. Springer, 2016.
27. D. v. d. Linden, I. Hadar, and A. Zamansky. On the requirement from practice for meaningful variability in visual notation. In *Enterprise, Business-Process and Information Systems Modeling - 18th International Conference, BPMDS 2017, 22nd International Conference, EMMSAD 2017, Held at CAiSE 2017, Essen, Germany, June 12-13, 2017, Proceedings*, 2017.
28. D. v. d. Linden, I. Hadar, and A. Zamansky. What practitioners really want: requirements for visual notations in conceptual modeling. *Software & Systems Modeling*, 2 2018.
29. D. v. d. Linden, A. Zamansky, and I. Hadar. How cognitively effective is a visual notation? On the inherent difficulty of operationalizing the physics of notations. In R. Schmidt, W. Guédria, I. Bider, and S. Guerreiro, editors, *Enterprise, Business-Process and Information Systems Modeling*, Cham, 2016. Springer.
30. D. v. d. Linden, A. Zamansky, and I. Hadar. A framework for improving the verifiability of visual notation design grounded in the physics of notations. In *2017 IEEE 25th International Requirements Engineering Conference (RE)*, Sept. 2017.