

Facilitating the adoption of standards through model-based representation

Morayo Adedjouma, Gabriel Pedroza, Asma Smaoui, Trung Dang

► **To cite this version:**

Morayo Adedjouma, Gabriel Pedroza, Asma Smaoui, Trung Dang. Facilitating the adoption of standards through model-based representation. ICECCS 2018 - 23rd International Conference on Engineering of Complex Computer Systems, Dec 2018, Melbourne, Australia. pp.213-216, 10.1109/ICECCS2018.2018.00035 . cea-03282263

HAL Id: cea-03282263

<https://hal-cea.archives-ouvertes.fr/cea-03282263>

Submitted on 8 Jul 2021

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.

Facilitating the Adoption of Standards through Model-based Representation

Morayo Adedjouma¹, Gabriel Pedroza¹, Asma Smaoui¹, and Trung Kien Dang²

¹Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France

²LIX, Ecole Polytechnique

{morayo.adedjouma,gabriel.pedroza,asma.smaoui}@cea.fr, trung-kien.dang@polytechnique.edu

Abstract—Nowadays standards provide recommendations for system development process in various phases and activities, aiming to address and reduce the risks associated to poor or flawed designs. Indeed, to increase systems assurance levels, a variety of concerns like safety, security and reliability are currently considered as critical and targeted by standards. However, to ensure such assurance levels, all stakeholders involved in the product development cycle (manufacturers, regulators, etc.) need to have a clear understanding of the standards contents. This paper presents an approach that aims to ease the common understanding of standards and their adoption by automatically generating models from them. The approach mostly relies upon the Business Process Model and Notation (BPMN) language. The potential benefits of this structured graphical view of standards are to facilitate their comprehension, visualization and navigation by the stakeholders involved in product development cycle.

I. INTRODUCTION

It is commonly agreed that limited, poor, misleading or wrong requirements specifications will lead to unacceptable levels of quality in final products. In particular, regarding safety-critical designs, as long as those requirements are finally fulfilled and depend on artifacts and functions deployed on-board (Software and Hardware), a malfunction may not only impact non-critical sub-systems (e.g., car infotainment, GPS) but also the critical ones (e.g., car powertrain, steering, braking). The malfunction of any safety-critical system may finally lead to personal injury. In the end, the economic costs at stake are due to loss of clients' trustiness, legal costs, and the final loss of product-makers marketplace. Legal, industry and research communities have conducted efforts in order to properly address referred risks. As outcomes, several recommendations and guidelines have been issued and published as standards which contain guidelines for ensuring process, product and services to develop safe, secure and reliable systems. However, the complexity of standards along with the large volume of documents produced in support of are often difficult to comprehend and difficult to use in their original format - which include text statements, tabular, graphical and other notation elements for illustrative purpose. For instance, the IEC 61508 standard, about functional safety of E/E/PE safety-critical systems [1], is written in seven parts on approximately 900 pages. Similarly, the ISO 26262 standard, about automotive functional safety [2], is written in 10 parts on approximately 450 pages. These documents also embed traceability information (using cross references) that are hidden by the volume and the complexity of their structure. These drawbacks induce many readability issues and obstruct having a precise and common understand-

ing of fundamental concepts by stakeholders participating in the systems production: engineers, safety specialists, business analysts, managers, regulators, etc. The goal of the this paper is to ease the adoption of standards in industry by allowing stakeholders to easily comprehend their structure, the various processes including the management and development life-cycle, and reveal the underlying traceability between these processes. The MDE approach can support meeting this goal. Indeed, as shown in recent studies, the Unified Modelling Language (UML) [3] has been adopted by several companies and organizations as the core language relying upon different MDE solutions. For those companies, the integration of standards into their development cycle can be achieved by modelling them. One candidate language to conduct standards modelling is the Business Process Model and Notation (BPMN) language [4] since it includes elements matching with most typical standards structures (mostly following an intertwined items pattern). Following a MDE perspective, this paper presents an approach for transforming the text-based standards into a graphical structured model for easy visualization and navigation while enabling assistance for advanced traceability analysis. Specifically, we make the following contributions: (1st.) the challenges in using the standards are highlighted and discussed in Section II; (2nd.) a generic approach proposed to address these challenges comes in Section III; (3rd.) the framework that implements the approach is built on MDE and Natural Languages Processing (NLP) tools [5] and is described in Section IV; (4th.) our approach is evaluated on the ISO 26262 standard in Section V; (5th.) a discussion on the approach strengths and limitations in meeting the challenges with standards comes in Section VI; and, finally, (6th.) a positioning regarding similar works is provided in Section VII.

II. STANDARDS USAGE AND ADOPTION CHALLENGES

Several challenges still exist to use standards in an industrial practice given their textual format. Some challenges are intrinsic to the nature and scope of the standard and cannot be easily overcome by a technical solution; targeting individual property of systems, terms ambiguities, vagueness in requirements, outdated recommended technologies, etc. Other drawbacks come from the standards complexity and are potentially resolvable by technical solutions; a large size along with an extensive number of cross references the standards often include may induce readability issues including misleading or imprecise comprehension of contents. Along with a complex cross referenced structure, the standards often include concepts with various syntaxes and semantics; a same term can have

different meaning in different standards, whereas different terms can be used for describing similar concepts. Standards provide guidelines and best practices in various engineering phases and activities. It happens that within the same activity, some recommendations address the development process while others prescribe methods, design goals, or architecture metrics for product development. In general, different stakeholders manage the process and product development requirements. Having such intertwined process-product based recommendations makes it difficult the deployment of the standard as, finally, each stakeholder may need to go through all clauses to identify the tasks related to their concerns. Besides, to assess standard conformity, the general practice is based on check listing over hundreds of required work products. Each work product in turn is evaluated as complete based upon the check-list of the unitary requirements that need to be satisfied for its production. The generated documentation may result excessive as long as many requirements do not directly impact product quality nor integrity whereas checking them out may result cumbersome, time-consuming and prone to human-error. Moreover, relying upon a check-list based completion process may lead to dismiss the thorough evaluation of requirements' fulfillment or to a weak evaluation of acceptance criteria which can undermine the essence of standard conformity.

III. APPROACH FOR AUTOMATED STANDARDS MODELING

The figure 1 shows an overview of our approach for automated transformation of standards from plain text into a model-based representation. The approach has been inspired from previous work about automated resolution of cross references in legal texts [6]. The approach has five sequential steps that

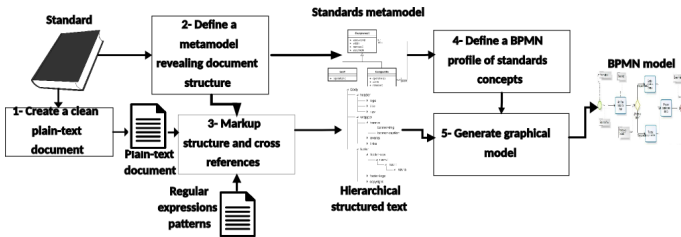


Fig. 1: 5-step approach overview

feature MDE and NLP paradigms. Most of the steps are automated, but some activities need to be manually performed. For instance, defining the metamodel of standards is mostly a human-based activity. *Step 1* concerns converting the standard from its original format - e.g. pdf format - into a plain-text document with an automated routine. The generated text file is cleaned up from the headers, footers, page number, etc. and any other decoration unnecessary for standard modeling. The graphical content is also extracted from the pdf document for future processing. In *Step 2*, we define a generic metamodel that captures fundamental concepts on the standards structure. In *Step 3* we define a BPMN profile where elements of the metamodel from *Step 2* map customized BPMN elements that correspond to concepts within the standard. We discuss *Step 2* and *Step 3* in subsection III-A. In *Step 4*, from the cleaned text file obtained in *Step 1* and the metamodel in *Step 2*, information are automatically extracted and structured yielding a tree-like structure file. The subactivities of this step include:

(1) Markup sections and subsections headings and their contents, and (2) Markup cross references. This step is discussed in Section III-B. The *Step 5* concerns using the outcomes of *Step 3* and *Step 4* to generate a BPMN model of the standard for visualization and navigation. The graphical model is built by applying a recursive algorithm which transforms items within the tree-like structure file, including cross references, into values that populate the target BPMN elements.

A. Defining a customized representation for standards

The figure 2 shows the metamodel of the standards in an UML Class diagram. The metamodel is built from a thorough examination of several standards, including the IEC 61508 and the ISO 26262, to ensure structure generality. However, others standards that do not fit the ISO/IEC pattern structure may need tailoring to adjust to their concepts. The IEC/ISO standards include several parts with a nested structure: a *Part* includes several *Clauses*, a *Clause* consists of one or many *Subclauses*, and particular *Subclauses*, namely *Input*, *Output*, point out to produced work products by the *Clause*. Composition links denote the containment relationships between the concepts. Other elements, such as *Figure*, *Table* and *Annex*, have an attribute named *path* used to store the location of some pictures or documents they refer to. Based

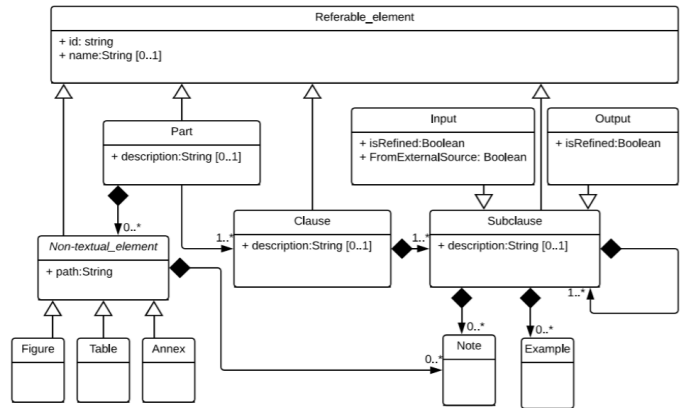


Fig. 2: Metamodel of standards textual body structure

on the metamodel in Figure 2, a mapping is settled from the concepts related to standard structure into the BPMN language. As BPMN is not specifically intended for standards modeling, the native BPMN concepts are specialized to better capture the standards structure concepts by introducing specific metamodel attributes. For the sake of simplicity and generality, we use a one-to-one mapping rule to define our specialized BPMN concepts mapping, as shown in Table I.

The figure 3 illustrates an excerpt of the mapping of concepts from ISO 26262, Part 5, into BPMN elements. Notice that the mapping preserves the hierarchical

Metamodel concepts	BPMN concepts
Part	Process
Clause, Subclause	Subprocess
Input	Data Input
Output	Data Output
Note, Example	Task
Table, Figure	Data Object
Annex	Call Activity

TABLE I: Mapping from standards structure into BPMN concepts

organization of the document. For example, when a *Process* or *Subprocess* is instantiated, all contained elements such as *Data Input*, *Task*, *Data Object*, etc. are also instantiated within the parent *Process* or *Subprocess*.

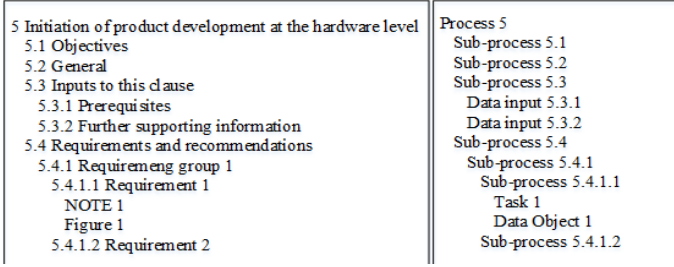


Fig. 3: Hierarchy in ISO 26262 (left) and in BPMN (right)

B. Extracting structured information from standards

The two activities comprising the extraction of structured information are described in the following items.

Markup sections. To markup the structure of the document, we use a similar approach to the Named Entity Recognition, a subcategory of information extraction [7]. First, named entities in the text are recognized and allocated into predefined groups such as names of sections, contents of sections. To do so, several pattern rules are defined and used to recognize the annotations over the context of a *Part*; *SectionHeading* for heading of the section, and *SectionContent* for contents of sections (see figure 4). Once recognized, annotated elements are identified relying upon an *uid* attribute which combines higher level and own element identifiers.

```

Rule: SectionHeading
(((SpaceToken.kind == control)):pre (((Token.kind != word)*):uid
{{Token}}{SpaceToken.kind == space})*:name {{SpaceToken.kind == control}}:pos)
--> mypattern.SectionHeading
  
```

Fig. 4: Rule for section heading recognition and annotation

Annotate cross references. A cross reference may exhibit single or multiple citations, both inside or outside the current document. To annotate cross references, a *Reference* pattern is defined to recognize every cross reference expression. The information extracted in the *Reference* annotation is used to support and ensure traceability between BPMN elements in the model of the standard. The main issue in this step is on resolving the Word Sense Disambiguation problem [8]; in some cases, many pattern rules are matching the same annotation and conflicts may occur. To resolve this ambiguity, a temporary set of annotations of fundamental grammars is proposed and applied in subsequent phases based upon a priority mechanism for matching the longest pattern.

IV. TOOL SUPPORT

We implement our approach as an Eclipse-based plug-in. The plug-in builds on the Gate NLP Workbench [9], the Papyrus modeling framework [10] and utilizes advanced technologies and languages such as UML, BPMN, JSON,

Java and some OpenCV libraries. Once a raw plain-text document is obtained, the GATE tool is used as a native pipeline information extractor. Several JAPE (Java Annotation Patterns Engine) scripts are developed and run in GATE to recognize regular expressions, identify the structure of the document and detect the cross references. The text metadata are extracted into a tree-like structure corresponding to the plain-text document. Then, JSON tool is used to capture this tree-like structure enriched with metadata. Finally, the framework communicates through GATE with Papyrus, an open-source tool based upon eclipse that provides an integrated environment supporting UML/BPMN modelling, to obtain the graph-like model. Papyrus is used as development platform for defining the ISO 26262 metamodel, which is turned into a profile as an extension of its native BPMN profile.

V. CASE STUDY

The evaluation on the case study aims at investigating the effectiveness and usefulness of our approach. To do so, we select the ISO 26262 standard from the perspective to process a very large volume of documents. The ISO 26262, about automotive functional safety aims to address the safety risks associated to poor or flawed designs for programmable systems. The standard is quite representative in terms of its size (10 parts, 450 pages), structure (86 clauses per-part) and contents (sections, subsections, references, tables, images). The clauses consist of several requirements whose fulfillment help to produce work products. We worked on transforming the technical requirements in ISO 26262. In particular, it is shown how to process Clause 5 and below of Part 3 to Part 7 of the standard. The structures of Clause 1 to Clause 4 are identical in all parts and they are about scope, references, definitions, etc. The figure 5 shows an excerpt of the BPMN model automatically generated from the pdf file of Part 5 of ISO 26262. The BPMN process model reveals the implicit

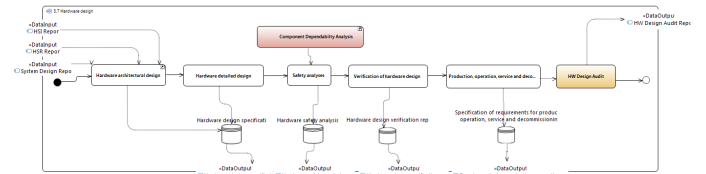


Fig. 5: Generated BPMN model from ISO 26262, Part 5

workflow underlying the development process with a graphical notation. The diagram shows the data which are conveyed through the different processes, origin and destination of work products, and specific outputs serving as evidence for the functional safety property. For example, the work product 7.5.1 *Hardware design specification* can straightforwardly visualized in the diagram being the result of both Subprocesses, namely, *Hardware Architecture design* and *Hardware detailed Design*. Hence, the BPMN diagram provides a structured graphical view to navigate inside the standard. More specifically, the standard clauses, represented by processes, include subclauses which are indeed sub-processes directly reachable from graphical elements. Regarding cross references, an advanced resolution mechanism solves implicit references by recognizing and pointing to concerned text elements or notions and by solving

multivalued references, e.g. a sentence pointing to a range of clauses: *Clause 7.4.1 to 7.4.5*. Any non-textual information in the document, such as tables, figures, formulas, etc. are also embedded within their respective clauses.

VI. DISCUSSION

Strengths. As a graphical language, a BPMN model can help engineers during the adoption of a standard or its integration into the engineering process in several ways. First, reading hundreds of pages may not be as effective in understanding the standard's scope, as an instant glance at the graphical model; processes encapsulate subprocesses, tasks and data flows and allow to highlight key terms. The navigability, traceability and sequentiality of elements in the diagram are ensured by the links that have been modeled. These features facilitate standard understanding by different stakeholders of the development process and improve effectiveness of interactions; stakeholders got a clear idea of which, when and how inputs and outputs are expected and by who. Another advantage is that process debugging and optimization can be carried out, for instance, to parallelize, merge or redistribute tasks and efforts in the aim of decreasing costs. This Papyrus-BPMN framework can be leveraged so as to assist process designers in a variety of activities like process monitoring through the development cycle and analysis of pertinence for the tasks-per-role distribution (management, designers, testers, etc.). Last but not least, the specialized framework inherits and benefits from Papyrus features which ensure its modularity and portability, and the interfacing with other Papyrus add-ons for systems design and concerns analysis (requirements, safety, security).

Limitations. The understanding of fine-grained elements of the BPMN model is at stake when their quantity is high and their location within the nested hierarchy is deep. A massive amount of references along with data flow links between tasks may lead to a crowded model, not easily readable (breakdown of graphical gain). As shown in figure 5, a partial view only including clauses and their links with work products is an option to prevent this effect from appearing. Then, a user-controlled visualization of clauses details, sub-clauses, cross references, etc. may help to partially tackle the issue, not without an impact on navigation effort. The BPMN model provides support for cross references to non-textual elements like tables, figures, etc. However, the referred elements are automatically extracted from the text-based standard and finally stored within the directory model as independent files. Since the model elements can only include local paths to the files, their access and usage is not straightforward; even if tables, figures and other non-textual elements may convey relevant information, accessing, searching and processing such information should be conducted manually by the user.

VII. RELATED WORK

Our approach is inspired from previous work on analysis of legal texts [6]. However, to the best of our knowledge, this work is the first attempt to provide a full framework that parses the standards and translates them into a graphical BPMN model to facilitate adoption. Whereas many works have proposed frameworks to ensure compliance with the ISO 26262 standard, e.g., [11], [12], referred approaches mostly target the

assurance process management, likely assuming intervention of standard experts. A similar work proposing standard modeling is the OpenCoss framework [13]: it provides means to share a non-ambiguous and formal interpretation of standards but the modeling activity is manually conducted what, being time-consuming and error-prone, does not ease adoption. Other works like [14] also propose standard modeling, mostly for certification purposes, and relying upon the SPEM standard. However, they also lack of automated modeling support and do not ensure full standard coverage.

VIII. CONCLUSIONS AND PERSPECTIVES

In this paper, we propose an approach to transform standards from their original plain text format into a graphical representation as a customized BPMN model in the aim to ease standard visualization, navigation and comprehension. We define the BPMN model by performing intermediary transformations: (1) from the textual-based document to a tree-like structure relying upon the JSON transducer, and (2) from the JSON tree-like structure to elements belonging to a specialized BPMN profile. The approach is applied to the ISO 26262 standard and the first outcomes from this case study show approach feasibility and effectiveness. Although measures have not been gathered on the time gained by using our framework versus manually modeling the standard, the model of each ISO 26262 part can be generated in a few seconds. Among the proposals to improve our approach, non-textual standard elements (figures, tables) should be formatted and processed in advance so as to avoid embedding them into the BPMN model. Integrating more advanced BPMN concepts (e.g., parallel gateways for concurrent tasks modeling) may help to encapsulate complex processes thus gaining in model abstraction, readability and comprehension.

REFERENCES

- [1] IEC, *IEC 61508: Functional safety of electrical/electronic/ programmable electronic safety-related systems*. IEC, 2010.
- [2] ISO, *ISO 26262: Road Vehicles : Functional Safety*, 2011.
- [3] OMG, *The Unified Modeling Language Specification Version 2.5*, 2015.
- [4] —, *Business Process Model And Notation Specification Version 2.0*, 2011.
- [5] N. Indurkha *et al.*, *Handbook of Natural Language Processing*, 2nd ed. Chapman & Hall/CRC, 2010.
- [6] M. Adedjouma *et al.*, "Automated detection and resolution of legal cross references: Approach and a study of luxembourg's legislation," in *(RE)*, 2014.
- [7] M.-F. Moens, *Information Extraction: Algorithms and Prospects in a Retrieval Context (The Information Retrieval Series)*. Springer, 2006.
- [8] E. Agirre *et al.*, *Word Sense Disambiguation: Algorithms and Applications*, 1st ed. Springer, 2007.
- [9] H. Cunningham *et al.*, *Developing Language Processing Components with GATE Version 8*. Gate, 2015. [Online]. Available: <http://gate.ac.uk>
- [10] Eclipse, *Papyrus modelling tool*, 2015. [Online]. Available: <https://www.eclipse.org/papyrus/>
- [11] M. K. S. Pandian *et al.*, "Analysis of iso 26262 compliant techniques for the automotive domain," in *MASE@MoDELS*, 2015.
- [12] M. Trei *et al.*, "An iso 26262 compliant design flow and tool for automotive multicore systems," in *PROFES*. Springer, 2016.
- [13] A. Ruiz *et al.*, "Systematic application of iso 26262 on a seoc: Support by applying a systematic reuse approach," in *(DATE)*, 2015.
- [14] B. Gallina *et al.*, "S-tunexspen: Towards an extension of spem 2.0 to model and exchange tunable safety-oriented processes," in *SERA*. Springer, 2014.