Enabling Automated Compliance Checking of Processes against Safety Standards

Julieth Patricia Castellanos Ardila

Innovation Design and Engineering - Mälardalen University

1 Problem and Research Question

Manufacturers of safety-critical systems have to comply with domain-specific safety standards to demonstrate that the deployment of their systems is acceptably safe. In some domains, the applicable safety standards establish the accepted procedures that regulate the development processes. Companies aiming at complying with such standards should adapt their practices and provide evidence that supports the fulfilment of the requirements. The production of this evidence, which is a mandatory input for the compliance assessment process, demands that the process engineer checks the fulfilment of every process-based requirement. This task is time-consuming and prone-to-error since standards are large documents with hundreds of process-related requirements that can be difficult to interpret. Also, a company can have many safety-critical-related processes to be scrutinized. The provision of automatic support for compliance can help organizations to have more control of their obligations as well as to add mechanisms to confirm the fulfilment of the standard's requirements. Thus, this thesis aims at providing an approach that facilitate compliance checking of the processes used to engineer safety-critical systems against the standards mandated (or recommended) in the safety-critical context. To address the overall research goal, we define concrete subgoals as follows:

1. Elicit the requirements to be met to support the automation of process-based compliance checking in the safety-critical context.
2. Identify methodologies that contribute to automate the compliance checking of planned process against process-based safety standards.
3. Facilitate the creation of formal specifications of the process-based requirements prescribed by safety standards.
4. Analyse existing methodological approaches that could be used for reusing compliance proofs in the safety-critical context.

2 Related Work

Automatic compliance checking of processes is a mechanism that provides benefits to compliance management [5]. In particular, researchers in the business context have exploited temporal logics for checking legal requirements [11,17,2]. However, the notions required for compliance are not adapted to facilitate the

of modelling standard's requirements. Approaches for checking safety requirements are presented in [12], in which the authors provide the steps for process reasoning and verification with Composition Tree Notations (CTN) and description logics (DL). In [13], BPMN is used to formally specify the software project, adopting the idea of model checking to enable detection and elimination of inconsistencies in process interaction. In [4], a validation of the process model is carried out with formal tools, specifically model-checkers available in the area of Petri nets. However, those approaches do not explicitly address the checking of process models against safety standards. In our approach, we have included the use modelling languages specially created for specifying development process, i.e., Systems & Software Process Engineering Metamodel (SPEM 2.0) [15] which may be preferred since they allow the creation of process method contents that can be reused in different kind of processes. SPEM 2.0-related community, to the best of our knowledge, has not addressed compliance checking. However, based on SPEM 2.0, some solutions for compliance management exists, such as the mapping of safety requirements to process elements [14,3]. The modelling of standards requirements is also exploited in [18] and used to detect whether the process model contains sufficient evidence for supporting the requirements. In our case, we have also exploited the modelling of standard requirements as well as the provision of a mechanism for including rules within the standard's requirements, which facilitate the resolution of the uncompliant situations that may result from the compliance checking.

3 Methods

Our research methodology, which was inspired in the research methodology for information systems research proposed in [16], consists of three main stages.

1. **Research Initiation:** Defines the overall research. In this stage, we *identify and motivate the problem* and *define the main goal*. The resources required in this stage include the knowledge of state of the art and the state of the practice. A problem formulation, which describes the main problem and formulates a motivation about the need to solve it, and an overall research goal, which is designed to address the main problem, are produced.
2. **Research Development:** Supports the achievement of the main goal. Initially, we *identify a sub-problem* and *define a subgoal*, which should describe a specific problem and justify the value of a solution. Later, we *design and develop* a solution artifact, i.e., constructs, models, methods, or instantiations, new properties of technical, social, and/or informational resources, that solves the specific problem. Within the artifact, its desired functionality, architecture and actual development have to be described. Then, the *demonstration*, which could involve the use of the artifact in experimentation, case study, proof or other appropriate activity, is carried out. These four steps are repeated for every research goal. Every iteration may finish in a global activity called *communication*, in which the problem and its importance, the

artifact, its utility and novelty, the rigour of its design, and its effectiveness is communicated to the research community and practitioners.
3. **Research Finalization:** Compile the project. We *integrate* the solutions of the subgoals and *validate* the overall research contribution, namely, we observe how well the artifact produced solves the overall problem.

4 Preliminary Results

Hitherto we have achieved some technical contributions. We have provided an automated compliance checking vision for the safety-critical context, in which process modeling and compliance checking capabilities are combined [8]. Within the vision, the set of elements required for creating process models checkable for compliance are identified and transformed into the specific language required by the selected compliance checker [9]. Safety compliance patterns [6] as well as methodological guidelines formulated for ISO 26262 [10] are also provided to facilitate the interpretation of safety requirements and its subsequent formalization. Finally, the design of a framework, which aims at planting the seeds for the future enablement of the systematic reuse of compliance proofs, is offered [7]. All the technical contributions have been demonstrated with academic examples.

5 Next Steps

To improve our results, we aim at exploiting process-line methodologies for enabling systematic reuse. Besides, we have considered the inclusion of process element beyond tasks for increasing the capabilities of the automated compliance checking vision. We also required to further validate the approach with more complex cases, i.e., industrial cases. Finally, we need to contemplate the use of patterns and methodological guidelines in more generalized ways to incorporate a wide range of standards and increase the applicability of the results provided by this thesis.

Acknowledgments. This work is supported by the EU and VINNOVA via the ECSEL JU project AMASS (No. 692474) [1].

References

1. AMASS.: Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems. http://www.amass-ecsel.eu/
2. Awad, A., Decker, G., Weske, M.: Efficient Compliance Checking Using BPMN-Q and Temporal Logic. International Conference on Business Process Management pp. 326–341 (2008)
3. Ayala, I., Gallina, B.: Towards tool-based security-informed safety oriented process line engineering. In: Proccedings of the 10th European Conference on Software Architecture Workshops (ECSAW). pp. 1–7 (2016)

4. Bendraou, R., Combemale, B., Crégut, X., Gervais, M.: Definition of an executable SPEM 2.0. In: 14th Asia-Pacific Software Engineering Conference. pp. 390–397 (2007)
5. Casanovas, P., González-Conejero, J., De Koker, L.: Legal Compliance by Design (LCbD) and through Design (LCtD): Preliminary Survey. In: 1st Workshop on Technologies for Regulatory Compliance Legal. pp. 33–49 (2017)
6. Castellanos Ardila, J.P., Gallina, B.: Formal Contract Logic Based Patterns for Facilitating Compliance Checking against ISO 26262. In: 1st Workshop on Technologies for Regulatory Compliance. pp. 65–72 (2017)
7. Castellanos Ardila, J.P., Gallina, B.: Towards Increased Efficiency and Confidence in Process Compliance. In: Stolfa J., Stolfa S., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2017. vol. 748, pp. 162–174. Springer, Cham (2017)
8. Castellanos Ardila, J.P., Gallina, B., Ul Muram, F.: Enabling Compliance Checking against Safety Standards from SPEM 2.0 Process Models. In: Euromicro Conference on Software Engineering and Advanced Applications (2018)
9. Castellanos Ardila, J.P., Gallina, B., UL Muram, F.: Transforming SPEM 2.0-compatible Process Models into Models Checkable for Compliance. In: 18th International SPICE Conference (2018)
10. Castellanos Ardila, J., Gallina, B., Governatori, G.: Lessons Learnt while formalizing ISO 26262 for Compliance Checking. In: Expected to be published in the procedings of Jurix (2018)
11. Elgammal, A., Turetken, O., van den Heuvel, W., Papazoglou, M.: Formalizing and applying compliance patterns for business process compliance. Software and Systems Modeling. pp. 119–146 (2016)
12. Kabaale, E., Wen, L., Wang, Z., Rout, T.: Representing Software Process in Description Logics: An Ontology Approach for Software Process Reasoning and Verification. In: Software Process Improvement and Capability Determination. SPICE 2016. Communications in Computer and Information Science. pp. 362–376 (2016)
13. Kingsbury, P., Windisch, A.: Modeling of Agile Avionics Software Development Processes through the Application of an Executable Process Framework. In: International Conference on Design and Modeling in Science, Education, and Technology (2011)
14. McIsaac, B.: IBM Rational Method Composer: Standards Mapping. Tech. rep., IBM Developer Works (2015)
15. Object Management Group Inc.: Software & Systems Process Engineering Meta-Model Specification. Version 2.0. OMG Std., Rev p. 236 (2008)
16. Peffers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems 24(3), 45–77 (2007)
17. Schumm, D., Turetken, O., Kokash, N., Elgammal, A., Leymann, F., van den Heuvel, W.: Business Process Compliance through Reusable Units of Compliant Processes. In: International Conference on Web Engineering. pp. 325–337 (2010)
18. Ul Muram, F., Gallina, B., Gomez Rodriguez, L.: Preventing Omission of Key Evidence Fallacy in Process-based Argumentations. In: 11th International Conference on the Quality of Information and Communications Technology (2018)