Towards Quantitative Evaluation of Reuse within Safety-oriented Process Lines

Barbara Gallina and Shankar Iyer

IDT, Mälardalen University
Box 883, 721 23 Västerås, Sweden
{barbara.gallina,shankar.iyer}@mdh.se

Abstract. Recently, Safety-oriented Process Line Engineering (SoPLE) has been proposed as a sound solution to systematize reuse in the context of safety-oriented processes described within safety-related standards. Currently, however, no metrics have been used to measure the actual gain in terms of reuse that the application of this engineering method entails. To overcome this lack of quantitative evidence, we adopt the GQM$^+$ Strategies model, an extension of the Goal/Question/Metric (GQM) paradigm, for measurements. After having defined our specific measurement goals, we build on top of existing metrics, defined for measuring product-related reuse, and we translate them in our semantic space to evaluate our goals. We then apply our GQM$^+$ Strategies model on a ECSS-compliant SoPL to illustrate and assess its usefulness.

Keywords: SoPLs, Process Improvement, Change management, GQM$^+$ Strategies, GQM, and ECSS-E-ST-40C.

1 Introduction

In the context of (safety) life-cycles mandated by standards, process improvement via Safety-oriented Process Line Engineering (SoPLE) seems to be feasible. SoPLE represents a solution to manage change in the context of safety-critical systems development processes. Change management is a key component in process improvement infrastructures [24]. Implementation of SoPLs in an organization, however, needs to be planned and justified. Metrics to measure their effectiveness have been so far neglected. A lack of metrics can impede their adoption. Organizations considering adoption of SoPLs are faced with the upfront questions regarding the selection of the right processes for conversion to derive the maximum benefits in the shortest time frame. Measurement of SoPLs starts at the design phase of software development and also covers all verification and validation activities.

Resources can be allocated to an endeavor only in the presence of objective justification of the economic benefits. To provide such justification, we need an appropriate measurement methodology. In this paper, we build on top of the

GQM$^+$ Strategies model [9] (shortened GQMPS), extension of the Goal/Question/Metric (GQM) paradigm [10], a goal-based software implementation and measurement paradigm. More specifically, we customize it to offer a SoPLE-targeted GQMPS, which we apply on a space domain-related SoPL. Finally, we discuss our findings and sketch future research directions.

The rest of this paper is organized as follows. In Section 2, we provide background information. In Section 3, we present our GQM$^+$ model for SoPL evaluation. In Section 4, we evaluate the benefits of an ECSS-compliant SoPL. In Section 5, we discuss the synergies between the SPI Manifesto and SoPLE-Targeted GQMPS. In Section 6, we discuss the related work. Finally, in Section 7, we conclude the paper and sketch future research directions.

2 Background

2.1 ECSS Standards: focus on software development

ECSS-E-ST-40C is a space standard that covers all aspects of space system software engineering including requirements definition, design, production, verification and validation, transfer, operations and maintenance [20, 3]. Tailoring rules are provided in a specific annex, *Annex R (normative)*, to enable manufacturers as well as suppliers to customize their engineering processes. The tailoring is conducted based on the software criticality, which ranges from A to D. Different customizations, performed by the different customers, can be seen as different single processes within a family of processes.

In this section, we limit our attention to a very small portion of ECSS-E-ST-40C, Section 5.6.2 (validation process implementation). We recall that this process is constituted of a series of activities. Each of these activities consists of various tasks, which in turn contain various steps.

Establishment of a software validation process (5.6.2.1)
- Establish validation process to validate the software (5.6.2.1a)
- Select associated methods, techniques, and tools for performing the validation tasks (5.6.2.1b)
- Determine the validation effort and the degree of organizational independence of the effort (5.6.2.1c)

Selection of an independent software verification and validation (ISVV) organization (5.6.2.2)
- Select a qualified organization if the project warrants an independent validation effort (5.6.2.2a)
- Assure the independence and authority of the conductor to perform the validation tasks (5.6.2.2b)

The activity *5.6.2.2* is composed of two tasks, *5.6.2.2a* and *5.6.2.2b*. According to Annex R, for instance, the tasks *5.6.2.2a* and *5.6.2.2b* are applicable for criticality levels A and B only. This limited process portion exemplifies what is typically required in terms of process engineering, i.e., complying with the requirements while tailoring.

2.2 SoPLE

SoPLE was introduced by Gallina et al. [14, 17] and applied in various domains (see for instance [15] for its usage in the context of tool qualification re-certification and [25] for its usage in the nuclear domain). SoPLE consists of a two-phase method. The first phase is aimed at engineering the domain from a process perspective i.e., identifying and systematizing process-related commonalities and variabilities in order to concurrently engineer a set of processes (SoPL). The second phase is aimed at deriving single safety-oriented processes via selection and composition of commonalities and variabilities. From a tooling perspective, SoPLE can be supported by the integration of Eclipse Process Framework (EPF) Composer [2] and Base Variability Resolution (BVR) Tool [1], where EPF Composer is used to model the base process and its related library, while BVR Tool is used to model (VSpec), resolve (Resolution) and realize (Realization) the variability. More precisely, VSpec permits users to model the variability in a feature diagram-like fashion. Resolution permits users to configure (make choices at variation points, where desired variants can be selected) their process. Finally, Realization permits users to bind the conceptual representation of the variable elements with the concrete elements in the base model. The integration of EPF Composer and BVR Tool is described in more details in [19].

2.3 Reuse-related Metrics

In [12], Berger et al. define several metrics which provide different perspectives for assessing the suitability for setting up product lines. Among such metrics, in this section, we recall Size of Commonality (SoC) and Product-related Reusability (PrR).

a) *SoC* measures the number of reusable/identical components in a product line. For evaluating a given set of similar products constituted of n products, each product pi with $1 \leq i \leq n, n \geq 2$ is decomposed into a set Cpi of m so called reasonable atomic pieces, where each piece is denoted cj with $1 \leq j \leq m, m \geq 1$. To perform a decomposition, all components/atomic pieces must be identified and formally specified. An annotated, directed graph is specified for each product which reflects the dependencies (called signatures) between all components. These dependencies can be logical or communicative. SoC is determined by comparing the component signatures. A syntactic comparison of signatures is made based on the names of the components while a semantic comparison is made from the behavioral profiles of the components which capture behavioral constraints. If the signatures are identical, the components are identical. *SoC* is computed as shown in Equation 1(a), where p_i represents the products of the product line, i ranges from *1* to n and C_{pi} represents the set of components of the product i.

b) *PrR* measures the extent of reusability of the common components for a specific product. *PrR* is computed as shown in Equation 1(b).

$$(a)\ SoC = \left| \bigcap_{1}^{n} C_{\mathrm{pi}} \right| \quad (b)\ P_{\mathrm{r}}R_{\mathrm{i}} = \frac{SoC}{|C_{\mathrm{pi}}|} \tag{1}$$

2.4 GQM$^+$ Strategies

The GQMPS model links measurement programs to higher level organization goals and strategies [9]. GQMPS is built as an extension of the GQM paradigm, a top down approach, in which measurements are based on measurement goals [10]. The GQM paradigm consists of three levels: the conceptual level (Measurement Goal) where the objectives are defined, the operational level (Question) where the questions are made and the quantitative level where the metrics are defined. These levels are also hierarchically organized in a pyramid structure. The apex of the pyramid is represented by a measurement goal, which specifies the purpose of measurement, the object which is being measured, the issue to be measured and the viewpoint from which the measurement is taken. This measurement goal is refined by a set of questions which breaks down the goal into its significant elements. Each question is further refined into one or more metrics. These metrics may either be objective or subjective in nature. Further, a particular metric may be used to answer more than one question.

The GQMPS model helps organizations to align multi-level organization goals and strategies to the measurement goals. It consists of two perspectives, the Organizational and Planning Perspective (OPP) and the Control Perspective (CP). The OPP and CP structures help incorporate dependencies among different levels of the organization. The OPP is concerned with the organizational goals and strategies while the CP is concerned with the measurements. The structure of the OPP resembles a pyramid with the top goal of the organization at the apex. The top goal is broken down into one or more strategies. Each strategy can be further split into one or more goals and associated strategies until the strategies cannot be further split into lower goals. The CP structure is built using the GQM paradigm. Each organizational goal is linked to a GQM structure in the CP via a measurement goal. These links enable alignment of organizational goals and strategies with measurement goals. This ensures that organizations invest resources only in meaningful and essential data collection and analysis activities.

3 GQM$^+$ Strategies Model for SoPLE evaluation

In this section we develop our GQMPS model for a SoPL with a focus on reusability of safety-oriented processes. A consolidated view of the GQMPS model is shown in Fig. 1. The SoPLE organization may be concerned with goals related to the feasibility of establishing SoPLs (pre-SoPL) or goals related to assessing the effectiveness of the established SoPLs (post-SoPL). In this paper, we limit our discussion to pre-SoPL goals, established SoPLs are not in place yet.

The OPP structure reflects the organizational goals and strategies starting with the overall organization goal at the top and the related strategies broken down below, reflecting the SoPLE organization goals and their related strategies. We show the association of the top organizational goal (*G1*) to the strategy (*S3*) of 'Exploit commonality of safety-oriented processes'. The strategy *S3* is further reduced to the software development organization (SDO) goal *G2*, that

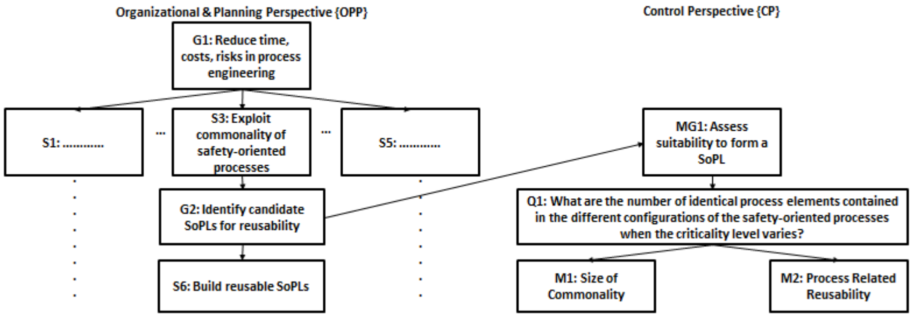

Fig. 1. SoPLE-targeted GQM+ Strategies Model

of 'identify candidate SoPLs for reusability' supporting the strategy *S6*, 'Build reusable SoPLs'. Strategy *S3* is reduced to a single SDO goal for illustration purposes, though, it may be reduced to additional goals, such as productivity and quality related goals. The CP part of the model links the goal *G2* to the measurement goal *MG1*, 'Assess suitability to form a SoPL'. The viewpoint is that of the SoPL manager who belongs to the SDO and has overall organizational responsibility for software development with formation of SoPLs as the object. *MG1* is progressively refined to questions *Q1* and *Q2* and the metrics *M1* and *M2* addressing the extent of commonality.

4 Evaluating the benefits of an ECSS-compliant SoPL

4.1 ECSS-compliant SoPL

Based on the information recalled in Section 2.1 and in Section 2.2, in this section, we model the ECSS-compliant SoPL related to the *Validation Process Implementation*. To model it, we use BVR/VSpec Editor. Fig. 2 presents our SoPL for *Validation Process Implementation (5.6.2)*. As mentioned in Section 2.1, the activity 5.6.2.2 is composed of two tasks. The variants of this activity depend on the associated process criticality. This results in a certain number of reusable elements and zero or more variable elements of the validation process implementation (5.6.2) depending on the associated criticality. The choices in our SoPL are the activities and tasks described in Section 2.1 and the criticality levels. One and only one of the four criticality levels are valid for any SoPL variant. The constraints are enforced when resolving a particular SoPL variant. For instance, in our SoPL, constraint c1 enforces the requirement that f7 is required only when the criticality level is either A or B.

4.2 Applying GQM+ Strategies

An application of our SoPLE-targeted GQMPS model is shown in Fig. 3. Goals *G1* and *G2* are the AMASS consortium organization goals [8] while *G7* is a

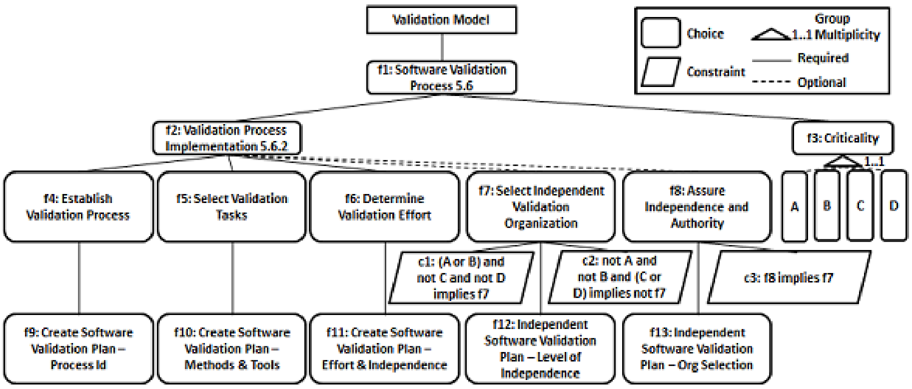

Fig. 2. ECSS-compliant SoPL

SoPLE organization goal, namely that of the Software Development Organization (SDO) of the space-related partner [5]. The pre-SoPL implementation goal of the SDO is *Identify candidate SoPLs for reusability*. The SDO goal *G7* links to the measurement goal *MG1*, *Assess suitability to form a SoPL*. Question *Q1* refines *MG1* and addresses process element counting based on the semantic signature of the processes subject to constraints imposed by criticality levels of the processes. *Q1* pertains to the commonality of the process elements. We can logically infer that, higher the commonality, the more likely is the suitability for adopting a process line approach. We define two metrics (*M1* and *M2*), regarding the question *Q1*, which are *SoC* and *PrR* respectively interpreted for processes i.e., *SoC* measures the number of reusable/identical components in a (safety-oriented) process line and *PrR* measures the extent of reusability of the common components for a specific process.

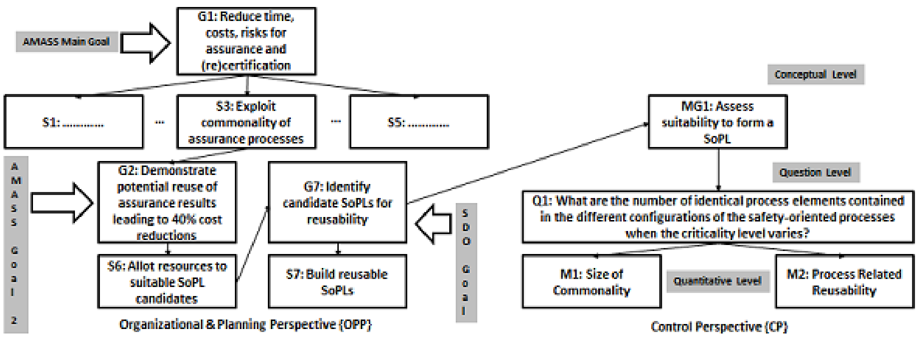

Fig. 3. Application of our SoPLE-targeted GQM$^+$ Strategies Model

Validation Process Implementation constitutes our SoPL and is treated in a similar manner to the product line referred to in Section 2.3. The activities which make up our SoPL constitute the elements and are similar to the components in the product line. The single processes can be configured based on the applicable criticality level and the associated constraints. As we have 6 common elements (see features f4, f5, f6, f9, f10 and f11), *SoC* computes to 6. The metric *PrR* is computed for each single process. As our SoPL consists of 4 criticality levels, we have at least 4 single processes. The number of elements in the single processes for criticality levels A, B, C and D are 10 (f4 through f13), 10 (f4 through f13), 6 (f4, f5, f6, f9, f10 and f11) and 6 (f4, f5, f6, f9, f10 and f11) respectively. Thus, *PrR's* for single processes A, B, C and D are computed as 0.6, 0.6, 1 and 1 respectively.

4.3 Discussion

In this section we discuss the findings related to the application of SoPLE-targeted GQMPS. The discussion covers the following three main bolded aspects.

General soundness - Despite the simplicity of the considered example, we can state that the application of SoPLE-targeted GQMPS for measuring SoPLE effectiveness is sound since it has the potential to generate objective justifications for its adoption. We have employed our SoPLE-targeted GQMPS model to measure SoPL effectiveness with reference to the AMASS consortium organization and the space-related partner. Our rather small SoPL and the computed metrics establish a case for the adoption of SoPLE to the ECSS software development process. The *SoC* determined for our SoPL indicates a high degree of commonality despite the variability introduced by the criticality levels. We observe from the computed *PrRs*, the extent of reusability of the common elements for SoPLs related to criticality levels C and D are higher than that for levels A and B.

Maturity SoPLE-targeted GQMPS's maturity varies with respect to the examination's angle. In terms of motivation and definition, SoPLE-targeted GQMPS is fairly mature. However, in terms of validation, it is still in its embryo stage. A true in-depth validation in industrial settings has not yet been carried out. We need to further examine our findings and the impact of the other process elements, which have a bearing on the safety-oriented process line such roles, tools employed, input/output work products. With respect to roles, as known, processes are nothing if not performed by people [22] and people skills are recognized to be crucial and need to be audited. According to ECSS-M-ST-10C [4], 5.2.1.2.e, for instance, the supplier shall demonstrate that the key personnel have the necessary qualification, skills and experience to perform the task for which they are allocated. Clearly, the necessary skills may vary based on the criticality of the tasks to be performed. With respect to tools employed, their reusability will be constrained by the existence of evidence in favour of their usage to automate a specific critical task. A tool qualified for a certain level will need to be re-qualified to be used to automate a task of a higher criticality level. With respect to input/output work products, in the context of SoPLE, only their syntactical reusability is expected to be considered. Their semantic content is instead expected to be addressed via product line engineering. In addition to the space domain, other domains could/should be explored. Within the AMASS project, SoPLE-targeted GQMPS has been included as part of the AMASS measurement framework [6] and thus additional validation data should be developed as part of the last iteration of the project, after application of the measurement framework.

Tool support-SoPLE-targeted GQMPS is not yet tool-supported. The computation of Size of Commonality and Product-related Reusability could be easily implemented as additional functionalities of BVR Tool. Similarly, an Eclipse-based editor could be developed to be able to edit SoPLE-targeted GQMPS applications.

5 SPI Manifesto and SoPLE-targeted GQMPS: synergies

SoPLE-targeted GQMPS enables the measurement of SoPLE effectiveness and thus permits process engineers to achieve an objective justification regarding the economic benefits related to the application of SoPLE as software process improvement strategy. SoPLE represents a solution to manage change in the context of safety-critical systems development processes. Change management is a key component in process improvement infrastructures [24].

As discussed in Section 4.3, when applying SoPLE, not only the breaking down of the work should be considered but all other relevant process elements, which may have an impact on SoPLE evaluation as well as on Software Process Improvement. Concerning roles, in addition to the technical skills (as per standards used in safety-critical software engineering), management skills [21, 22] as well as social responsibility-related skills [23] should also be considered in order to ensure SPI-readiness attitude. Moreover, as observed by Gallina et al. [16] in the Swedish context, culture may also influence the SPI-readiness. Thus, technical skills alone are not sufficient and thus they cannot be analysed in isolation when reasoning about the reusability of a role from one process to a different one since cultural, managerial, and social responsibility-related (lack of) skills may (hinder/) foster his/her successful deployment. Existing dependencies between skills and other process elements should be specified. To this purpose, the integration between EPF Composer and BVR Tool offers an adequate tool support for SoPLE. Via EPF Composer, process engineers have the possibility to model a single process of the SoPL and then make it vary via BVR Tool (more specifically, via the VSpec, Resolution, and Realization editors). As shown in Fig. 2, process engineers have also the possibility to specify constraints that limit the selection and composition of reusable process elements. Skill-related constraints could/should be specified in order to properly constrain the role-reusability based on contextual information.

6 Related Work

In the literature, other works have tackled the necessity of measuring reuse. In [18], Her et al. define a framework for evaluating reusability of core assets in a product line. The authors model the measures based on various characteristics of core assets. A key limitation of the defined metrics is their subjective nature. Our GQMPS model also considers metrics based on core asset characteristics but the characteristics can be judged objectively. Moreover, we also relate measurement objectives to organization goals and strategies. In [13], Dijkman et al. propose similarity measures for comparison of process models held in a repository to enable reuse, based on node matching, structural and behavioral similarity with similar results. We also make use of similarity-based measures but we build on top of different metrics. Becker et al. [11] present a survey of methods to define and calculate 23 similarity measures for business processes and compare these when applied to a same set of models to study the differences in similarity and observe differences based on process characteristics. Our work focuses on safety-oriented processes and not on business processes. Moreover, our primary goal in this paper is to shape a methodological framework for measurement.

7 Conclusion and Future Work

In the context of safety life-cycles mandated by standards, process improvement via SoPLE seems to be feasible. Implementation of SoPLs in an organization, however, needs to be planned and justified. This requires the ability to measure the effectiveness of the implementation at all stages of the development effort. In this paper, we have proposed a SoPL-targeted GQMPS model to enable the measurement of SoPLE effectiveness. Then, we have applied it and demonstrated the effectiveness of SoPLE in the context of ECSS standards. We may logically infer that because we measured and have the evidence for appropriateness of SoPLE, implementation should enable process improvement. Our SoPL is built on a very small part of the ECSS specifications and adopts an organization goal/strategy linked measurement model. Though our study has been performed in relation to an organization's specific processes and the associated standards, we examined only pre-SoPL implementation metrics.

In the near term future, the proposed SoPL-targeted GQMPS model requires to be validated by assessing several SoPLs. Moreover, further work is required to extend our SoPLE-targeted GQMPS model to incorporate post-SoPL implementation metrics and link the related measurement goal to goal *G2* in our SoPLE-targeted GQMPS model. Finally, in a long-term future, tool-support will be developed.

Acknowledgments. This work is supported by EU and VINNOVA via the ECSEL JU under grant agreement No 692474, AMASS project [7].

References

1. BVR Tool. `http://modelbased.net/tools/bvr-tool/`. (Last accessed: September 7, 2018).
2. Eclipse Process Framework Project. `http://www.eclipse.org/epf/`. (Last accessed: September 7, 2018).
3. ECSS, ECSS-E-ST-40C, Space Engineering–Software. `http://wwwis.win.tue.nl/2R690/doc/ECSS-E-ST-40C(6March2009).pdf`. (Last accessed: September 7, 2018).
4. ECSS, ECSS-M-ST-10C, Space project management - Project planning and implementation. `http://everyspec.com/ESA/download.php?spec=ECSS-M-ST-10C.048180.pdf`. (Last accessed: September 7, 2018).
5. AMASS. Deliverable D1.1. Case Studies Description and Business Impact, 2017. (Last accessed: September 7, 2018).
6. AMASS. Deliverable D1.3. Evaluation framework and quality metrics, 2017. (Last accessed: September 7, 2018).
7. AMASS (Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems). `http://www.amass-ecsel.eu`. (Last accessed: September 7, 2018).
8. AMASS Consortium: Objectives. AMASS. `http://www.amass-ecsel.eu/content/objectives`. (Last accessed: September 7, 2018).
9. V. Basili, A. Trendowicz, M. Kowalczyk, J. Heidrich, C. Seaman, J. Münch, and D. Rombach. GQM+ Strategies in a Nutshell. In *Aligning Organizations Through Measurement*, pages 9–17. Springer, 2014.
10. V. R. Basili, G. Caldiera, and H. D. Rombach. The Goal Question Metric Approach, 1994. *Citado*, 3:11, 2012.
11. M. Becker and R. Laue. A comparative survey of business process similarity measures. *Computers in Industry*, 63(2):148–167, 2012.
12. C. Berger, H. Rendel, and B. Rumpe. Measuring the Ability to Form a Product Line from Existing Products. *arXiv preprint arXiv:1409.6583*, 2014.
13. R. Dijkman, M. Dumas, B. Van Dongen, R. Käärik, and J. Mendling. Similarity of Business Process Models: Metrics and Evaluation. *Information Systems*, 2011.
14. B. Gallina, S. Kashiyarandi, H. Martin, and R. Bramberger. Modeling a Safety-and Automotive-oriented Process Line to Enable Reuse and Flexible Process Derivation. In *IEEE 38th International Computer Software and Applications Conference Workshops (COMPSACW)*, pages 504–509, 2014.
15. B. Gallina, S. Kashiyarandi, K. Zugsbrati, and A. Geven. Enabling cross-domain reuse of tool qualification certification artefacts. In *1st International Workshop on DEvelopment, Verification and VAlidation of cRiTical Systems*, volume 8696 of *Lecture Notes in Computer Science*, pages 255–266. Springer, 2014.
16. B. Gallina and M. Nyberg. Reconciling the iso 26262-compliant and the agile documentation management in the swedish context. In *Critical Automotive applications: Robustness & Safety (CARS), Matthieu Roy, Paris, France, HAL*, September 2015.
17. B. Gallina, I. Sljivo, and O. Jaradat. Towards a Safety-oriented Process Line for Enabling Reuse in Safety Critical Systems Development and Certification. In *35th Annual IEEE Software Engineering Workshop (SEW)*, pages 148–157, 2012.
18. J. S. Her, J. H. Kim, S. H. Oh, S. Y. Rhew, and S. D. Kim. A Framework for Evaluating Reusability of Core Asset in Product Line Engineering. *Information and Software Technology*, 49(7):740–760, 2007.

19. M. A. Javed and B. Gallina. Safety-oriented Process Line Engineering via Seamless Integration between EPF Composer and BVR Tool. In *22nd International Systems and Software Product Line Conference (SPLC), Sept 10-14, Gothenburg, Sweden, in press.* ACM Digital Library, 2018.
20. M. Jones, E. Gomez, A. Mantineo, and U. K. Mortensen. Introducing ECSS Software-Engineering Standards within ESA. `http://www.esa.int/esapub/bulletin/bullet111/chapter21_bul111.pdf`, August 2002.
21. M. Korsaa, M. Biro, R. Messnarz, J. Johansen, D. Vohwinkel, R. Nevalainen, and T. Schweigert. The SPI Manifesto and the ECQA SPI manager certification scheme. *Journal of Software: Evolution and Process*, 25(5):525–540, 2012.
22. M. Korsaa, J. Johansen, T. Schweigert, D. Vohwinkel, R. Messnarz, R. Nevalainen, and M. Biro. The people aspects in modern process improvement management approaches. *Journal of Software: Evolution and Process*, 25(4):381–391, 2013.
23. R. Messnarz, M. A. Sicilia, M. Biro, E. G. Garcia Barriocanal, M. Rubio, K. Siakas, and A. Clarke. Social responsibility aspects supporting the success of SPI. *Journal of Software: Evolution and Process*, 26(3):284–294, 2014.
24. Pries-Heje, J., Johansen, J. (eds.). MANIFESTO Software Process Improvement eurospi.net, Alcala, Spain (2010). `http://www.iscn.com/Images/SPI_Manifesto_A.1.2.2010.pdf`. (Last accessed: September 7, 2018).
25. T. Varkoi, T. Mäkinen, B. Galllina, F. Cameron, and R. Nevalainen. Towards systematic compliance evaluation using safety-oriented process lines and evidence mapping. In *24th European & Asian Systems, Software & Service Process Improvement & Innovation (EuroAsiaSPI2), Ostrava, Czech Republic, 5-8 September.* Communications in Computer and Information Science, vol 748. Springer, Cham, pp. 83-95, 2017.