On the Explicit Consideration of Context Variability in the SPES Modeling Framework

André Heuer, Tobias Kaufmann

paluno – The Ruhr-Institute for Software Technology
University of Duisburg-Essen
Gerlingstr. 16
45127 Essen - Germany
{andre.heuer|tobias.kaufmann}@paluno.uni-due.de

Mihail Constantinescu-Fomino

Airbus Defence and Space
Rechlinger Straße
85077 Manching - Germany
mihail.constantinescu-fomino@cassidian.com

Abstract: The extended SPES Modeling Framework (SPES MF (ext.)) supports the development of software for embedded systems and recognizes variability in a dedicated perspective. Variability was introduced to the SPES MF to consider different customer needs and to enable the efficient engineering of multiple product variants of software for embedded systems. Hence, an explicit variability model documents the differences of product variants, addressing different customer needs. Following the notion of the SPES MF (ext.), coarse grained engineering problems are decomposed into more fine grained engineering problems. This decomposition requires distinguishing between the current engineering subject and the elements in the context of that engineering subject, which cannot be altered. Based on the experience made in several applications domains (e.g. avionics), we were able to distinguish context variability and non-context variability. As well as elements that are in the context of an engineering subject, context-variability cannot be altered. Because context variability affects the variability of the product variants under development, it needs to be explicitly considered during engineering. Today, context variability is not made explicit in a variability model. Therefore, we propose to recognize context variability in the variability perspective of the SPES MF (ext.) as a separate view. We demonstrate the applicability of the proposed approach by an example taken from the avionics domain.

1 Introduction

The SPES MF (ext.) explicitly recognizes variability in terms of a variability perspective [HKW13]. Thereby, the increasing need of many application domains (e.g. in the automotive and avionics domain) to explicitly manage variants of embedded software has been recognized. Today various stakeholders like customers, users or national authorities for legislation demand for stakeholder specific solutions resulting in manifold and potentially mutually exclusive stakeholder needs. These needs are addressed by variable embedded software that can be efficiently derived to meet the stakeholders demand. In order to engineer such variable embedded software, the variability of the embedded software needs to be explicitly considered and documented with respect to the

Acknowledgement
This paper was partially funded by the BMBF project SPES 2020_XTCore under grant 01IS12005C.

engineering artefacts that are created during the engineering process. Hence, an explicit and continuous management of the different variants of the embedded software throughout the engineering process is required. In [HKW13] and [KMW14], the authors propose to define a variability perspective orthogonal to the existing SPES viewpoints based on role-specific concerns. This perspective allows for documenting variability and its relation to the artefacts defined by the SPES MF. In contrast to the artefacts in the SPES MF, the variability perspective does not differentiate between different levels of granularity.

In [HP14], the authors distinguish between system variability and context variability. In contrast to system variability, context variability is situated in the context of the engineering subject. Because context variability is a property of particular elements in the relevant context of variable embedded software, it cannot be altered, changed or modified by the engineers focusing on a specific engineering subject. This also implies that the binding of context variability is not in control of these engineers. Therefore, it is required to explicitly differentiate between system variability and context variability for an engineering subject. However, the differentiation between context and system variability is not explicitly considered in the SPES MF, neither in the variability perspective nor in the context model of the requirements viewpoint. Because of SPES MF's "divide and conquer" principle, engineering subjects (e.g. subsystem, component), system variability and context variability is specific to a granularity level and the focused engineering subject. Therefore, we propose using views based on first class concerns to address the granularity level-specific differentiation of context and system variability in the variability perspective of the SPES MF (ext.).

The paper is structured as follows: In Section 2, foundations of the SPES MF, variability modelling in the SPES MF and the definition of role-specific views are described. Section 3 introduces the definition of different views for context and system variability onto the SPES MF (ext.) variability perspective. Section 4 describes an application example from the avionics domain. In Section 5, related work is presented. A summary and a conclusion are presented in Section 6.

2 Foundations

The SPES Modelling Framework. The SPES Modelling Framework (cf. [Br12]) allows for structuring modelling artefacts during the engineering of software for embedded systems. It consists of four viewpoints constituting the base layer: The *Requirements Viewpoint* addresses the structured documentation and analysis of requirements. The *Functional Viewpoint* addresses the structured documentation and analysis of system functions. The *Logical Viewpoint* addresses the structured documentation and analysis of the logical solution, whereas the *Technical Viewpoint* addresses the structured documentation and analysis of the technical solution. All four viewpoints cover multiple levels of granularity, which can be individually defined according to the needs of the engineering process. A new granularity level is created, whenever a coarse-grained engineering artefact is decomposed into multiple finer grained engineering artefacts. For each of the engineering artefacts the four viewpoints are applied to ensure a structured engineering path on the lower granularity levels. In addition, the SPES MF explicitly considers crosscutting system properties (e.g. safety, real-time).

Variability Modelling for the SPES Modelling Framework. In [HKW13], the SPES MF was extended by the *variability perspective* to consider the crosscutting nature of variability constituting the extended SPES Modelling Framework. The variability perspective documents the variability information orthogonally to the existing SPES viewpoints in a single variability model. This variability perspective can be understood as an instantiation of the concept *perspective* proposed by ROZANSKI and WOODS [RW12] and defines the ontological concepts for variability modelling in the SPES MF (ext.). In the variability perspective of the SPES MF (ext.), the orthogonal variability model (OVM) can be used for modelling differences between product variants in terms of a variability model (cf. [PBL05]). The OVM uses the concepts of *variation point* and *variant*. A *variation point* is defined as a variable item of the real world or a variable property of such an item, e.g. the paint of a car. The set of all variation points in an OVM are denoted as the set $VARIATIONPOINTS$. A *variant* is defined as a particular instance of a variation point, e.g. red paint (cf. [PBL05]). The set $VARIANTS$ is the total set of all variants in an OVM. A valid product defined by an OVM is constituted of a selection of specific variants. Variation points and variants are interconnected by either an optional or a mandatory variability dependency which are subsumed in the set $DEPENDENCIES$. An optional variability dependency can also express alternative groups. Additionally, constraints between variation points and variants are used to express *requires*- or *excludes*- relationships. Those constraints are denoted by the set $CONSTRAINTS$. Variants in an OVM can be related to artefacts in the base layer of the SPES MF. For instance a variant can be related to a goal in a goal model in the requirements viewpoint. This goal is only part of a product if its related variant was selected for this specific product.

Defining Views on Models. Views are based on concerns of different roles participating in the engineering process. As the SPES MF does not provide an engineering process, we use the V-Modell XT (cf. [V09]) as the V-Modell XT is a well-established software process model. Relevant concepts for defining views and their relations are shown in Figure 1. In the engineering of variable embedded software multiple different *process roles* (short: roles) are active (cf. [V09]). Each of these roles has a *responsibility* assigned.

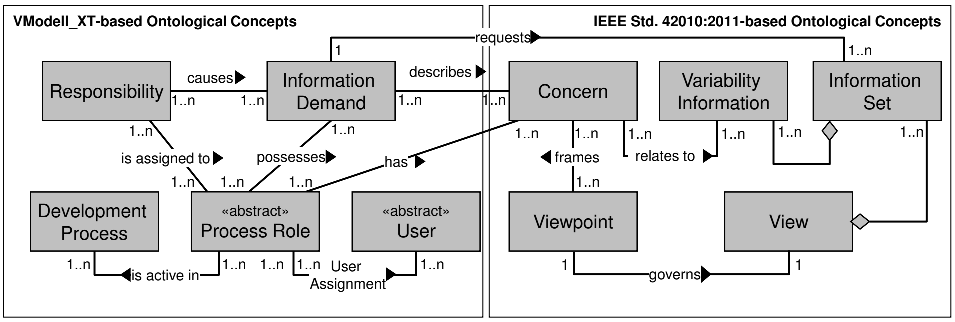

Figure 1: Conceptual Relations between process roles and IEEE 42010 concepts

Moreover, each role is assigned to a concrete *user*. These responsibilities are understood as the reason for an information demand that is possessed by a concrete process role. An information demand is the reason for requesting a specific *information set*. Furthermore, an *information demand* describes one or more *concerns*. One or more concerns frame a *viewpoint*, which governs a *view*. Hence a viewpoint is understood as a specification for a view. A view is composed of one or more *information sets*. Each information set comprises a certain subset of *variability information*. Those variability information are explicitly documented in a variability model. This information corresponds to the concerns and fulfils role-specific information demands.

3 Defining Context and System Variability as Views

The variability perspective as proposed in [HKW13] does not distinguish between system and context variability as described in [HP14]. Therefore, we propose to specify a *context variability view* and a *system variability view* in the variability perspective of the extended SPES MF. Views on instances of variability models focus on specific variability-related concerns, which are documented in the variability model. Hence, views allow for analysing a concern in isolation to get a deep understanding of this concern (cf. [GJM03]). This notion led us to introduce variability viewpoints based on the core concepts of IEEE Std. 42010: stakeholder-specific *concern* and *viewpoint* as a specification for a *view* (cf. [III11], Section 2). A variability *view* can thus be understood as a role-specific excerpt of information from the variability perspective. A *variability viewpoint* that addresses a role-specific variability-concern specifies which information of the variability perspective is needed by the corresponding role to be able to fulfil its responsibilities. Essentially, variability viewpoints document role-based projections, inspired by relational database theory (cf. [EN11]), on variability information. Thus, concerns are the conceptual fundament for specifying variability viewpoints. We explicitly relate the concerns of roles to system and context variability viewpoints and thereby allow a role-based structuring of the variability perspective. This structure can be independent from the viewpoint in the SPES MF and their corresponding concerns.

3.1 Relevant Concerns for Context and System Variability View

Concerns related to Context Variability. In order to explicitly manage variability in context, variability information need to be identifiable as context variability information. Following our ontological understanding (cf. Section 2), we propose relating variability information to specific concerns expressing key interests in variability information. Such concerns are related to a specific process role that can be named context variability engineer. The set of concerns relevant to this role can be characterized as follows:

- Which variability information of a variability model cannot be manipulated for the current engineering subject? **(C1)**
- Which context elements in the relevant context of the current engineering subject are variable? **(C2)**
- Which effect does the variability in context have on the system for the current **(C3)**

engineering subject?

We understand context variability as a proper subset of the total amount of variability information documented in the variability perspective, which is in a relation to the concerns C1, C2, und C3. These concerns are elements of the set *CONCERNS*.

Concerns related to System Variability. The system variability view comprises the variability information that is related to the engineering subject. Hence, this variability information is the variability information that can be manipulated during engineering of the engineering subject. From this understanding we derive the concerns C4 and C6.

- Which variability information of a variability model can be manipulated for the current engineering subject? **(C4)**
- Which elements of the current engineering subject are variable? **(C5)**
- Which elements of the current engineering subject are related to variable elements in the relevant context? **(C6)**

Following the preceding paragraph, the concerns C4, C5, and C6 are as well elements of the set CONCERNS. The concern C4 describes the elements of the variability model that can be altered by a role having a system variability view. In contrast, the concern C5 refers to the variable elements of a specific base artefact of SPES MF. Hence, such elements are related to the variability model. Variability information that is solely related to the concerns C4 or C5 is not affected by variability in context. To express the dependence of certain variable elements of an engineering subject to context variability, we propose concern C6. Thus, system variability that is solely related to the concern C6**,** documents the impact of the context variability onto the engineering subject.

3.2 Concern-specific Viewpoint Specification for Variability Information

Based on the two kinds of concerns proposed in the preceding section, we define the Context Variability Viewpoint and the System Variability Viewpoint. Both viewpoints are applied to the variability model (e.g. documented by an OVM or feature models) in the variability perspective. Both viewpoints define means to derive granularity level-specific views either for the role of a Context Variability Engineer or System Variability Engineer. While the context variability view includes variability information that is in the relevant context of the engineering subject (i.e. based on concerns C1-C3 see Section 3.1), the system variability view includes variability information of the engineering subject (i.e. based on concerns C4-C6, cf. Section 3.1). Based on the viewpoint specifications, different analysis can be conducted, e.g. the identification of context variability, incorrectly specified variability, or the identification of conflicts between context and system variability.

3.3 Granularity Level-based Views

In Section 2, we described the concept of different granularity levels in the SPES MF. As context variability as well as system variability depends on the granularity level, we propose to define granularity level specific views on variability models to structure the variability information. To do so, we propose a specification of operations on the aforementioned sets using the specification language Z (cf. [Sp92]). We relate the elements of LEVELS to the concerns described in Section 3.1. These level-concern pairs are then related to the elements of the variability model (cf. Figure 2, *Assignment_Spec)*. Based on this understanding, we propose a specification for a function that assigns levels, concerns and variability information in the aforementioned way (cf. Figure 2, *AssignLevelsAndConcerns_Spec*). To retrieve the variability information for a specific level-concern pair, we propose the specification of a level-specific variability information retrieval function (cf. Figure 2, *Create_View_Spec*). This function returns the variability information according to a given level-concern pair.

AssignLevelsAndConcerns_Spec

ΔAssignment_Spec
concerns?: $\mathbb{P}_1$*CONCERNS*
levels?: $\mathbb{P}_1$*LEVELS*
vInfos?: $\mathbb{P}_1$*VARIABILITYINFORMATION*

((levels?, concerns?), vInfos?) ∉ A_Level_Concern_VINFO ⇒ A_Level_Concern_VINFO = A_Level_Concern_VINFO ∪ {levels? ↦ concerns? ↦ vInfos?}

Create_View_Spec

ΞAssignment_Spec
level?: $\mathbb{P}_1$*LEVELS*
concern?: $\mathbb{P}_1$*CONCERNS*
result!: $\mathbb{P}_1$*VARIABILITYINFORMATION*

result! = A_Level_Concern_VINFO (level? ↦ concern?)

Assignment_Spec

A_Level_Concern_VINFO: $\mathbb{P}_1$ *((*$\mathbb{P}_1$*LEVELS ×* $\mathbb{P}_1$*CONCERNS) ×* $\mathbb{P}_1$*VARIABILITYINFORMATION)*

Figure 2: Operations to assign concerns to variability information and create views

The function specification described in Figure 2 can be further refined to express the assignment of concerns to specific elements of $VARIABILITYINFORMATION$ (e.g. level-concern pairs to variation point assignment etc.). Therefore, level-concern assignments specific to the sets defined in Section 2 can be specified. For instance, the concern to variation point assignment can be expressed as $A_{VP} \subseteq \big((LAYER \times CONCERNS) \times VARIATIONPOINTS\big)$. The same assignment strategy can be applied to the other sets describing particular elements of a variability model.

In order to establish the relation of concerns and process roles (cf. Figure 1), we denote the set of all possible process roles as $ROLES$. Hence, the set $ROLES$ includes both the process roles context variability engineer and system variability engineer (cf. Section 3.2). Therefore, we describe the concern to role assignment as $CA_{ROLES} \subseteq CONCERNS \times ROLES$ (e.g. $CA_{ROLES} = \{(C_1, ContextVariabilityEngineer), \ldots, (C_4, SystemVariabilityEngineer)\}$.

4 Application Example

We apply the proposed approach in the engineering of an Unmanned Aerial Vehicle (UAV) developed by Airbus Defence & Space. A pilot in the ground station controls the UAV. It is equipped with an optronics system allowing for surveillance operations. Based on the customer of the UAV, different optronics systems from different vendors (vendor A or vendor B) are built-in. Both vendors use different communication protocols and thus different interfaces, even if the physical interface is identical. The optronics system of vendor A (optronics A), for example, additionally comprises a laser designator. Beside these differences in the equipment of the optronics systems, both systems have different basic modes. Thus, supporting both optronics systems requires several adaptations in different systems of the UAV. In the following, the application example focuses on the Mission Payload & Management System (MPMS) that is responsible for the communication of the aircraft systems and the payload, i.e. optronics systems.

Figure 3 shows an example of the specification of the UAV software documented by the SPES MF (ext.). In the upper part, the context model (requirements viewpoint) and the logical architecture (logical viewpoint) are shown on the granularity levels system and subsystem level (or short: system level, subsystem level). The lower part of Figure 3 depicts the variability model in the variability perspective. Based on the sets defined in Section 3.3, the set VARIABILITYINFORMATION consists of the elements {("Vendor A, Vendor B", "1,1", "Optronics"), ("Adapters A, Adapters B", "1,1", "Adapters"), "Vendor A", "Vendor B", "Adapter A", "Adapter B", "Vendors", "Adapters"}, and the set LEVELS = {System, Subsystem}.

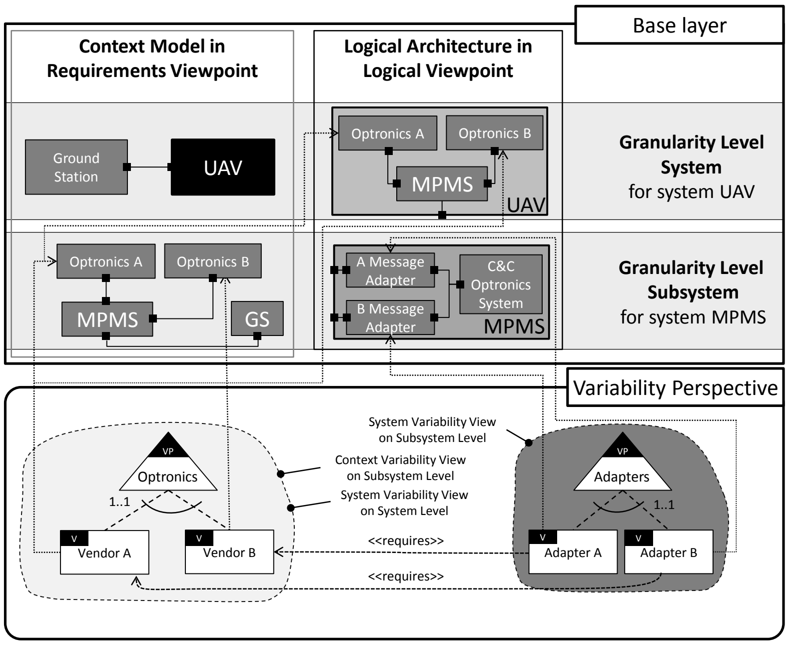

Figure 3: Context and system variability view between different levels of granularity

On system level, the ground station is in the context of the UAV as the pilot controls the UAV from the ground station. Data from the optronics system are transmitted to the ground station. On this level, the UAV is decomposed in several logical systems. The logical architecture in Figure 3 shows only a simplified excerpt of the logical architecture. It shows the MPMS that manages different payloads of the UAV. Thus, it is interfaced to the optronics A and optronics B. These optronics are variable as shown by the relation between the OVM in the lower part of the figure and the optronics A and B. As both optronics are variable in the system that is engineered on this level, the variation point and variant are system variability on the system level (concerns: C4, C5; function call: $AssignLevelsAndConcerns$ $(\{C4, C5\}, \{System\}, \{(\text{“}Vendor\ A, Vendor\ B\text{”}, \text{“}1{,}1\text{”}, \text{“}Optronics\text{”})\})$, cf. Section 3.3).

On the shown subsystem level, the MPMS system is modelled. The context model on the subsystem level shows the context of the MPMS including the optronics and the ground station (GS). The context model shows the different optronics systems in the context of the MPMS. As both systems are in the context, the variation point and the variant corresponding to the different optronics are now context variability on the subsystem level for the MPMS (concerns: C1, C2; function call $AssignLevelsAndConcerns(\{C1, C2\}, \{Subsystem\}, \{(\text{“}Vendor\ A, Vendor\ B\text{”}, \text{“}1{,}1\text{”}, \text{“}Optronics\text{”})\})$, cf. Section 3.3). The logical architecture of the MPMS comprises, amongst others, the Command & Control Optronics Systems (C&C Optronics System) and message adapters. As the MPMS has to interpret vendor specific messages as given by the optronics systems, those messages have to be converted for the C&C Optronics Systems. As two vendors for the optronics are defined by the context of the MPMS, two message adapters are realised in the logical architecture of the MPMS. These adapters are system variability of the MPMS, because they are engineered during the development of the MPMS (C4 and C5). Additionally, the adapters of the MPMS are related to variable elements in the context of the MPMS, i.e. optronics A and B (C6), resulting in the following function call: $AssignLevelsAndConcerns$ $(\{C4, C5, C6\}, \{Subsystem\}, \{(\text{“}Adapter\ A, Adapter\ B\text{”}, \text{“}1{,}1\text{”}, \text{“}Adapters\text{”})\})$. However, if the adapter for vendor A is selected, the optronics of the vendor B in the context is also required for a valid product configuration. The same hold true for the adapter for the optronics of vendor B. Thus, the variants of the adapters have a *requires* relation to the respective optronics (concern: C3; function call: $AssignLevelsAndConcerns$ $(\{C3\}, \{Subsystem\}, \{\text{“}Adapter\ A, Adapter\ B\text{”}\})$, cf. Section 3.3). Based on the concerns and level the $SystemVariabilityView_{System}$ can be derived by calling a function $Create_{View(\{System\},\{C4,C5\})}$ (cf. Section 3.3). This function call results in the variability information subset $\{(\text{“}Vendor\ A, Vendor\ B\text{”}, \text{“}1{,}1\text{”}, \text{“}Optronics\text{”})\}$. Based on this approach, granularity level-specific views can be derived for the variability perspective as part of the SPES MF (ext.).

5 Related Work

There are already exist approaches to model context variability. In [LK10] and [KL13], the authors analyzed the relationship of the environment to a product family and stated that the variability of the context has an impact on the goals and attributes of a family of systems and thus on the binding of variability for a product variant. They propose to model the variability of the context by means of feature models. Also HARTMANN AND TREW [HT08] and UBAYASHI ET AL. [UNH12] recognized the necessity to document context variability information and its relation to system variability and also propose to use feature models for modelling context variable. In [HT08], relations between context and system variability are explicitly modelled by requires and excludes constraints. TUN

ET AL. [Th09] augments the Jackson-Zave framework by variability information. System variability is documented by a feature model for each description in the framework, i.e., requirements, problem world context, and specification. However, system and context variability is not explicitly differentiated even for feature models of the problem world context. The above approaches allow considering context variability. But only few approaches (e.g. [HT08]) describe the relationship between context and system variability. Additionally, all approaches describe context and system variability only on a single level of granularity. Our approach also explicitly supports context and system variability on different levels of granularity and defines respective relationships.

Two different approaches to establish variability viewpoints are known in the literature. First, *augmentative* approaches augment certain elements of variability models with additional information, which is the foundation for view derivation. The approach in [SLW12] proposes feature models consisting of attributed features. These attributes are used to derive different views. Second, *descriptive* approaches specify sets of variability information by enumerating the elements that are part of a specific view. FEY et al. [FFB02] use feature sets to group features based on the needs of domain experts. In [Hu13] views on feature models based on feature enumerations are suggested. A set based approach to structure multi-dimensional product lines is proposed in [TH03]. In contrast to our approach, this approach does not explicitly consider role-specific variability-concerns. We understand levels and concerns as a first class concept. Based on this understanding, we relate levels to concerns. These pairs are then related variability information to existing elements. Consequently, we understand our approach as being *annotative* and hence not part of the aforementioned categories.

6 Conclusion

The variability model in the variability perspective does not support the differentiation between context and system variability, neither on a single level of granularity nor between different levels of granularity. Therefore, we proposed to use views based on IEEE Std. 42010 defined for context and system variability. The definition of views is based on role-specific concerns. We formally defined the views using a Z specification. We illustrated the applicability of the approach by a simplified UAV with variable optronics. For each engineering subject on each granularity level, the sets of context variability and system variability can be determined by the concerns. However, the union of both sets on a specific granularity level (e.g. subsystem level) is not necessarily equal to the set of system variability on a higher level of granularity (e.g. system level), because the engineering subjects on the, e.g. subsystem level are not necessarily in the context of each other. Thus, none of the concerns C1 to C3 applies. Also variability that is on lower granularity levels of other engineering subjects may not be relevant and are thus neither in the context variability view nor in the system variability view.

Our future work is to detail the Z specification of the views to allow an automated approach for analyzing the variability perspective, which is considered a first step towards automatic consistency checks. Additionally, we plan to further evaluate our approach on several granularity levels and with different engineering subjects on a granularity level.

References

[Br12] Broy, M. et al.: Model-Based Engineering of Embedded Systems – The SPES 2020 Methodology, Springer, Berlin, Heidelberg, pp. 31-48, 2012

[EN11] Elmasri, R.; Navathe, S.: Fundamentals of database systems. Boston: Addison-Wesley, 2011

[FFB02] Fey, D.; Fajta, R.; Boros, A.: Feature modeling: A meta-model to enhance usability and usefulness. In: Proc. 2nd SPLC 2002, San Diego, 2002. Springer, Lecture Notes in Computer Science, 2002; pp. 198-216

[GJM03] Ghezzi, C.; Jazayeri, M.; Mandrioli, D.: Fundamentals of software engineering. Upper Saddle River, N.J: Prentice Hall, pp. 44-49, 2003

[HKW13] Heuer, A.; Kaufmann, T.; Weyer, T.: Extending an IEEE 42010-Compliant Viewpoint-Based Engineering-Framework for Embedded Systems to Support Variant Management. In: Proc. 4th IESS 2013, Springer, Heidelberg, pp. 283-292, 2013.

[HP14] Heuer, A.; Pohl, K: Structuring variability in the context of embedded systems during software engineering. In: Proc. VaMoS'14, ACM, New York, 2014.

[HT08] Hartmann, H.; Trew, T.: Using Feature diagrams with Context Variability to model Multiple Product Lines for Software Supply Chains. In: Proc. 12th Int. SPLC 2008. IEEE Computer Society, 2008

[Hu13] Hubaux A. et al.: Supporting multiple perspectives in feature-based configuration. In: Journal of Softw. & Syst. Modeling, Volume 12, Issue 3, Springer, New York, 2013

[III11] ISO/IEC/IEEE Systems and Software Engineering – Architecture Description. ISO/IEC/IEEE Standard 42010:2011, 2011

[KL13] Kang, K.C.; Lee, H.: Variability Modeling. In: Systems and Software Variability Management: Concepts, Tools and Experiences. Springer, 2013.

[KMW14] Kaufmann, T.; Manz, C.; Weyer, T.: Extending the SPES Modeling Framework for Supporting Role-specific Variant Management in the Engineering Process of Embedded Software In: Software Engineering (Workshops) 2014, pp. 77-86, 2014.

[LK10] Lee, K.; Kang, K. C.: Usage context as key driver for feature selection: Springer. In: Software Product Lines: Going Beyond, pp. 32-46, 2010.

[PBL05] Pohl, K., Böckle, G., van der Linden, F.: Software Product Line Engineering: Foundations, Principles and Techniques. Springer, Berlin, Heidelberg, 2005.

[Po10] Pohl, K.: Requirements Engineering – Fundaments, Principles, and Techniques. Springer, 2010.

[RW12] Rozanski, N., Woods, E.: Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives. 2nd Edition, Addison-Wesley, Upper Saddle River, 2012.

[SLW12] Schroeter J.; Lochau M.; Winkelmann T.: Multi-Perspectives on Feature Models. In: Proc. MODELS 2012, Innsbruck 2012. Springer Berlin, Heidelberg, 2012; pp. 252-268

[Sp92] Spivey, J. The Z Notation: A Reference Manual. Prentice Hall, New York, 1992

[TH03] Thompson, J. M., Heimdahl M.P.E.: Structuring product family requirements for n-dimensional and hierarchical product lines. In: RE Journal, Volume 8, Issue 1, Springer, Berlin, Heidelberg, 2003; pp. 42-54.

[Th09] Than Tun, T.; Boucher, Q.; Classen, A.; Hubaux, A.; Heymans, P.: Relating requirements and feature configurations: A systematic approach. In: Proc. 13th SPLC 2009, pp. 201-210, 2009

[UNH12] Ubayashi, N.; Nakajima, S.; Hirayama, M.: Context-dependent product line engineering with lightweight formal approaches. In: Science of Computer Programming 78, no. 12, 2012.

[V09] V-Modell ® XT, Version 1.3, February 2009. http://v-modell.iabg.de/