Impact of Feature Interaction on the Safety Analysis for Unmanned Avionics Product Lines

André L. de Oliveira, Rosana T. V. Braga, Paulo Cesar Masiero, Ibrahim Habli, Tim Kelly

▶ **To cite this version:**

André L. de Oliveira, Rosana T. V. Braga, Paulo Cesar Masiero, Ibrahim Habli, Tim Kelly. Impact of Feature Interaction on the Safety Analysis for Unmanned Avionics Product Lines. Safecomp 2013 FastAbstract, Sep 2013, Toulouse, France. pp.NC. hal-00930892

HAL Id: hal-00930892
https://hal.science/hal-00930892

Submitted on 14 Jan 2014

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.

Impact of Feature Interaction on the Safety Analysis for Unmanned Avionics Product Lines

André L. de Oliveira[12], Rosana T. V. Braga[1], Paulo C. Masiero[1], Ibrahim Habli[2], Tim Kelly[2]
[1]Mathematics and Computer Science Institute, University of São Paulo, São Carlos-SP, Brazil
[2]Department of Computer Science, University of York, Deramore Lane, York, United Kingdom
{andre_luiz, rtvb, masiero}@icmc.usp.br, {ibrahim.habli, tim.kelly}@york.ac.uk

***Abstract*—Unmanned Avionics Systems (UAS) are real-time critical embedded systems that include high-integrity requirements. Most of these systems need to be certified before use, particularly in civil airspace. To reduce development cost, some UAS software is developed as part of a Software Product Line (SPL). A product-line comprises a reference architecture and a set of reusable core assets. New systems can be derived from the product-line architecture and core assets based on a predefined process that manages and controls permitted variations, based in part on product-line features defined in a feature model. However, many features are interdependent and hence complicate the analysis of all potential feature combinations for product-line systems. In this paper we discuss the impact of feature dependencies in the safety analysis of unmanned avionics SPLs and present a preliminary model-based solution for managing the impact of these dependencies.**

Keywords—unmanned avionics; product-lines; feature interaction;

I. INTRODUCTION

Unmanned aircraft systems are systems built to support aircrafts that do not require a human pilot [1]. Software Product Lines (SPL) consist of systems that share a common set of core requirements that differ according to a set of allowable variations [2]. SPL have been used to develop avionics software [3]. Safety-critical product-lines, such as avionics systems, include high-integrity safety requirements. For avionics SPLs to be used, it is necessary that both systems and aircraft be certified against pre-established guidance. Variability analysis and management is crucial for development of safety-critical SPLs, for which it should be considered in both product-line development and safety analysis. For example, safety case development [4] is an approach that has been used for documenting assurance arguments for safety-critical systems, such as avionics, in order to obtain certification credit. As with other product-line assets [2], product-line safety cases need to include mechanisms for managing the impact of variation [5]. Establishing a balance between safety assurance and reuse management is a challenging task in product-line safety analysis because a safety-critical SPL should satisfy its safety properties in all products derived from selecting and combining product-line features and assets.

II. RESEARCH PROBLEM

Feature interaction is defined as a feature or features affecting the behaviors of some other feature(s) [6]. This affects product-line safety analysis and development assets in both SPL domain engineering and application engineering. Thus, the following questions arise: a) how can product line safety/hazard analysis address feature interaction for avionics? b) How can assurance be provided that product-line assets are ready for reuse in several allowable configurations? Here we focus on two main challenges: 1) certification: certification authorities typically deal with *single* product certification and not with a product-line; and 2) feature interaction: dealing with the dependence relationships between product-line assets and how to provide assurance for the reuse of both product-line development and safety analysis assets. Feature interaction variation in product-line development assets and their operational environment (usage context) have a significant impact on safety analysis assets related to hazard identification, risk analysis, risk management (mitigation measures), risk monitoring, risk acceptance, and safety case argumentation. The addition of a feature into a safety-critical product-line can potentially lead to changes in many safety analysis assets, because it is necessary to consider and analyze the interaction of the new feature with other SPL features to perform safety analysis.

Variability management problems in avionics safety-critical SPLs relate to the complex traceability between functional dependencies (in aircraft functions) and product-line feature interaction and among product-line development, safety analysis, and safety assurance (safety cases) assets [4][5]. There are proposals for metamodels in the literature that address some of these problems, as the product-line Functional Failure Model [7] and the OMG Structured Assurance Case Metamodel [8], but there is little guidance, methods, or techniques that describe how to use such models together to manage such traceability. There is also no automated tool-support to use these models in order to analyze the traceability between product-line development, safety analysis, and safety argumentation assets.

III. PROPOSED SOLUTION

Our proposed solution to deal with feature interaction problems in avionics SPL development and safety assessment assets is based on the concept of 'problem-solution feature interaction' and a feature interaction mapping approach built based on this concept [6]. Problem-solution feature interaction is defined as an interaction between two or more architectural/implementation features (solution-space) that only arises based on one or more domain features (problem-space) from the feature model. The feature interaction mapping approach proposed by Sanen et al. [6] combines concepts of configuration knowledge and feature interaction, and the provisioning of automated tool support for complex mappings. In safety-critical SPLs, safety knowledge should be part of SPL configuration knowledge. Such knowledge covers SPL safety assets such as hazard logs, risk assessment, mitigation measures and argumentation data. So, in order to reuse knowledge about certain hazards and conditions in safety-critical SPLs we should incorporate 'safety knowledge' into 'configuration knowledge'.

We can abstract the 'problem-solution feature interaction' concept to address the traceability problem in safety-critical product lines. For example, avionics product-line feature models (domain models) map to the problem-space part of the concept, while Functional Dependency Models from avionics software can map to the solution-space part. We can also extend the concept of 'problem-solution feature interaction' to address safety (i.e.

variation interaction in safety assessment data) in safety-critical product lines. The reason for this is that there is also interaction between product-line development and safety assets. Thus, in the same way that the concept of 'problem-solution feature interaction' we can have interactions between one or more safety requirements (in the safety domain) that only arise in the presence of one or more feature (requirements/architectural) interactions in a specific usage context. To support modeling of feature interactions in SPLs, languages such as Feature-Oriented Requirements Modeling Language (FORML) [9] can be used. In this language, SPL modeling considers two viewpoints: the world problem, which comprises domain modeling using feature models to specify valid SPL combinations, and behavior model, to model feature interaction considering each SPL feature separately (feature module) using state-machines. FORML can be used to express feature interaction relationships in SPL feature and avionics functional models; and for expressing safety requirements interaction in safety-critical SPLs.

In order to address safety-critical product-line traceability for the UAS domain, we firstly propose a mapping between SPL feature interactions and avionics system function dependencies using merging metamodels, interfaces and parsing techniques. Model merging is a process of merging two source models 'M_A' and 'M_B', instances of 'MM_A' (feature interaction) and 'MM_B' (functional dependence) metamodels, into a target model 'M_C', which is an instance of 'MM_C' (merged) metamodel. We aim to use a merging language, such as Epsilon Merging Language (EML) [10], to build our proposed merged metamodel for feature interaction and avionics functional dependencies, within the Eclipse Modeling Framework (EMF). EMF will be used to provide automated tool support for traceability between product-line development, safety analysis, and safety argumentation assets.

We also propose other traceability merging metamodels to map core and variation points in product-line development (feature and context models), safety analysis (hazard identification, risk analysis, risk management), safety argumentation (safety cases) assets, manned and unmanned aircraft certification guidance, and product-line processes [2]. The merging metamodel for the safety case will be built based on the Goal Structuring Notation (GSN) [4][5] and the OMG SACM metamodel [8] and integrated with product-line processes [2]. From using our proposed merging metamodels, it will be possible to get traceability between a product-line feature associated with one specific usage context, and its correspondent safety analysis data, such as hazards related to the features in the assumed context, risk analysis data as risk severity and probability of occurrence, risk mitigation measures to be adopted, risk acceptability analysis, and safety argumentation (safety case models).

The presence of such traceability can contribute towards improvements in providing assurance of product-line features and feature interaction safety properties. This can be justified because the use of these metamodels can improve the management of product-line feature interaction safety requirements. We believe the use of such approach can facilitate the certification process of product-line configurations (through easier identification and management of dependencies) by reusing pre-certified safety analysis and safety argumentation data. The use of our merging metamodels can also contribute to reduce the complexity of adding new features and feature interactions to an existing product-line, due to the traceability between product-line interactions, safety analysis, and safety argumentation assets. To support and facilitate the use of our proposed metamodels, we are developing a UAS product-line development process and guidance to support the management of avionics software development, safety analysis and argumentation activities and their assets. Ongoing work involves developing tool support for both metamodels and the UAS development process, and validating the metamodels in real world case studies.

IV. RELATED WORK

Product line safety has been addressed in the literature, e.g. in Liu et al.[11], Habli et al. [7], and the MISSA Project [12]. Liu et al. [11] integrated SPL safety analysis with model-based development in a state-based modeling approach using two product-line safety analysis techniques: Software Failure Modes, Effects and Criticality Analysis, and Software Fault Tree Analysis. The MISSA Project [12] proposed a solution for assigning DALs for avionics systems developed from a set of models, by using Functional Dependency Models (FDM) and safety analysis tools. FDM is used for decomposing functions (features in an SPL) into sub-functions that correspond to classes or levels, or functional failure modes that impact the effects of a function failure condition. After all functional failure modes and all classes of functional performance are found, the decomposition is closed by allocating physical resources to implement the function. This data is processed by safety analysis tools to support the allocation of DALs to functions and their possible combinations. Habli et al. [7] proposed an SPL functional hazard model which is integrated with product-line context and domain (feature) models, and a model-based SPL hazard assessment approach aimed at integrating functional hazard assessment to product-line domain engineering and application engineering phases.

ACKNOWLEDGEMENTS

CNPq Brazilian research agency (grant 152693/2011-4).

REFERENCES

[1] J. P. Potocki de Montalk, "Computer software in civil aircraft', in: Proceedings of 6th annual conference on computer assurance, systems integrity, software safety and process security, 1991, pp. 10-16.

[2] P. Clements, L. Northrop, Software product lines: practices and patterns, Addison-Wesley Professional, 3a ed., 2002.

[3] F. Dordowsky, R. Bridges, H. Tschope, Implementing a software product line for a complex avionics system, In: 15th SPLC Conference, 2011, 241-250.

[4] T. Kelly, A systematic approach to safety case management, in: SAE world congress, Society for Automotive Engineers, 2003.

[5] I. Habli, T. Kelly, A safety case approach to assuring configurable architectures of safety-critical product lines, In: 1st ISARCS, Springer-Verlag Berlin, Heidelberg, 2010, 142-160.

[6] F. Sanen, E. Truyen, W. Joosen. 2009. Mapping problem-space to solution-space features: a feature interaction approach. In *Proc.*GPCE, ACM, 167-176.

[7] I. Habli, T. Kelly, R. Paige. Functional Hazard Assessment in Product-Lines: A Model-Based Approach. In *Model-Driven Product-Line Eenginering*, 2009.

[8] OMG, Structured Assurance Case Metamodel (SACM), available on-line: http://www.omg.org/spec/SACM.

[9] P., Shaker, J. M., Atlee, S. Wang, "A feature-oriented requirements modelling language," *Requirements Engineering Conference,* v. 151, n. 160, 24-28, 2012.

[10] D. S. Kolovos, R. F. Paige, F. A. C. Polack. Merging models with the epsilon merging language (EML). In *9th* MoDELS, Springer-Verlag, 215-229, 2006.

[11] J. Liu, J. Dehlinger, R. Lutz. Safety analysis of software product lines using state-based modeling, J. of Systems and Software, v80, n11, 1879-1892, 2007.

[12] MISSA Project, MISSA Project Final Report: Extract of the Publishable Summary, More Integrated Systems Safety Assessment, 2011.