REVaMP 2 Project: Towards Round-Trip Engineering of Software Product Lines -Approach, Intermediate Results and Challenges

Andrey Sadovykh, Tewfik Ziadi, Alessandra Bagnato, Thorsten Berger, Jan-Philipp Steghöfer, Jacques Robin, Raúl Mazo, Elena Gallego

REVaMP² Project: Towards Round-Trip Engineering of Software Product Lines - Approach, Intermediate Results and Challenges

Andrey Sadovykh[1,3], Tewfik Ziadi [2], Alessandra Bagnato[1],Thorsten Berger[4], Jan-Philipp Steghöfer[4], Jacques Robin[5], Raul Mazo[5,6], Elena Gallego[7]

[1] Softeam, Research & Development Department, France
andrey.sadovykh@softeam.fr, alessandra.bagnato@softeam.fr

[2] Sorbonne University, Paris, France
tewfik.ziadi@lip6.fr

[3] Innopolis University, Russia
a.sadovykh@innopolis.ru

[4] Chalmers | University of Gothenburg
thorsten.berger@chalmers.se, jan-philipp.steghofer@gu.se

[5] University Paris 1 Panthéon-Sorbonne
{jacques.robin, raul.mazo}@univ-paris1.fr

[6] GIDITIC, Universidad Eafit, Medellin, Colombia
rimazop@eafit.edu.co

[7] The REUSE Company, Spain
elena.gallego@reusecompany.com

Abstract. The REVaMP² Project is a major European effort towards Round-Trip Engineering of Software Product Lines for software intensive systems. Indeed, software is predominant in almost every modern industry. The importance of time-to-market has grown tremendously in many business domains. Organizations are in a constant search for approaches for mass production of highly customizable systems. The software product lines engineering approach promises to provide up to 10x speed increase benefits in time-to-market. Traditionally, automated tools proposed a top-down approach, i.e., variants were generated from a model of the product line. However, the industry used a bottom-up approach that helped to re-create a product line out of various clones of a system. This operation is very costly and error prone. The goal of REVaMP² is to automate the process of extracting a product line from various system artifacts and help with verification and the co-evolution of the product line. The project involves 27 partners that contribute with diverse research and industrial practices to address case study challenges stemming from 11 application domains. In this paper, we would like to present the motivation for the project, the current approach, the intermediate results and challenges.

Keywords: Software Product Lines Engineering, Round-Trip Engineering, Model-Driven Engineering, Extraction, Co-evolution, Verification, Tools, ITEA3.

1 Introduction

An ever-higher proportion of B2B and B2C products and services acquire leading market positions by becoming more software-intensive. This trend is illustrated by buildings and vehicles evolving from electro-mechanical systems into Cyber-Physical Systems (CPS) and by services such as utilities and transportation evolving towards personalized, adaptive offers based on analytics of data generated by the Internet of Things (IoT). This technological trend reinforces with the shift away from traditional product sales towards service subscription packages, which include leasing a product as one item in a customized turn-key service offer. These Software-Intensive Systems and Services (SIS) create and adapt to innovative market disruptions and customers' whims far quicker and at a lower cost than their less software based competitors. However, they also raise new engineering challenges. In particular, they require more agile, round-trip engineering processes that better leverage legacy assets, as well as a more systematic and automated variability management. An engineering process is called round-trip when it combines top-down steps that refine abstract assets such as requirement specifications and high-level architectural patterns into more concrete ones such as executable simulation models and source code, with bottom-up steps that abstract such these more concrete assets into the more abstract ones. Variability management refers to a method to systematically (a) reuse common assets shared by a whole family (or line) of system (or product or service) variants on a common theme and (b) organize and relate distinct assets proper to each variant along commercially and technologically relevant characteristics and constraints.

In this paper, we first summarize the main variability management challenges that SIS engineering companies face today, given the current State-of-the-Art (SotA thereafter), when they attempt to round-trip engineer SIS families at optimal cost by reusing legacy artifacts from past assets from their product or service portfolio. We then overview the current status and different outstanding challenges of the REVaMP[2] - Round-trip Engineering for VAriability Management Platform and Process project [1]. This is a collaborative research and innovation project labeled by the Eureka program ITEA-3 in a consortium of 27 partners in 5 European countries.

2 Motivation, Concept and Approach

Product Line Engineering (PLE) is a mature paradigm for variability management. It enables defining a family of product configurations to satisfy different customer needs and to later systematically generate the associated product variants by combining predefined reusable components. Benefits of PLE include achieving large-scale productivity gains and improving time-to-market and product quality. Reports describe gains following PLE adoption by as much as tenfold in productivity and quality, cost reduction by as much as 60%, decrease labour needs by as much as 87%, and decrease time to market (new variants) by as much as 98% [2]. As all sorts of devices, systems and services become more software intensive, the more they can benefit from PLE adoption. Commercially successful implementations of the PLE paradigm can be found in companies from domains ranging from avionics and automotive software, to printers, mobile phones or web applications.

However, adopting a PLE approach is still a major challenge and represents a risk for a company[3],[4], [5],[6]. First, compared to single-system development, PLE variability management implies a methodology that highly impacts the life cycle of the products as well as the processes and roles inside the company. Second, adopting PLE from the beginning, an approach called proactive PLE [5], is a subject to two main assumptions: 1) the company must have, in advance, a complete understanding of the variability to anticipate all possible variations; 2) the company should start from scratch to specify the variability and implement the reusable assets.

Berger et al. showed in a survey with industrial companies that participated in industrial PLE, that around 50% of them cannot adopt proactive PLE [7]. On the one hand, the variability in these companies is discovered as customer needs emerge over time; so, it is very difficult if not impossible, to anticipate all the variations from the beginning. On the other hand, companies already have existing product variants that were implemented using an opportunistic reuse in an ad-hoc way to quickly respond to different customer needs. As mentioned by Dubinsky et al. [8], instead of adopting PLE, many companies clone an existing product and modify it to fit the new customer needs. This approach, called clone-and-own, is widely used because it is initially faster to start with an already developed and tested set of assets [8].

Fig. 1 illustrates, the three main PLE processes: proactive, extractive and round-trip. Proactive PLE is shown on the left of the figure. It must start with the inception of the project in a high-cost upfront investment step t0 called domain modelling. During this phase, the requirements for the entire product line must be simultaneously elicited. From the resulting PL, all product variants satisfying the variability model constraints can then be automatically generated in a second step t1. In Fig. 1, the domain model mandatory features are grey squares, the variant-specific features are coloured squares, and constraints on features mutual exclusivity are annotated with the XOR operator. An extractive PLE is illustrated on the right of Fig. 1. It starts by the rapid development of a Minimal Viable Product (MVP). If this MVP fits its market, it is then followed by sequentially and opportunistically cloning-and-owning variants to quickly target other niches for which many common features from the initial product can be reused (steps t1 to t4). When these variants and the constraints among them become too numerous to be efficiently managed without an explicit and systematic variability model, they are then refactored and consolidated in bottom-up fashion into a PL (t5). Round-trip PLE combine both approaches.

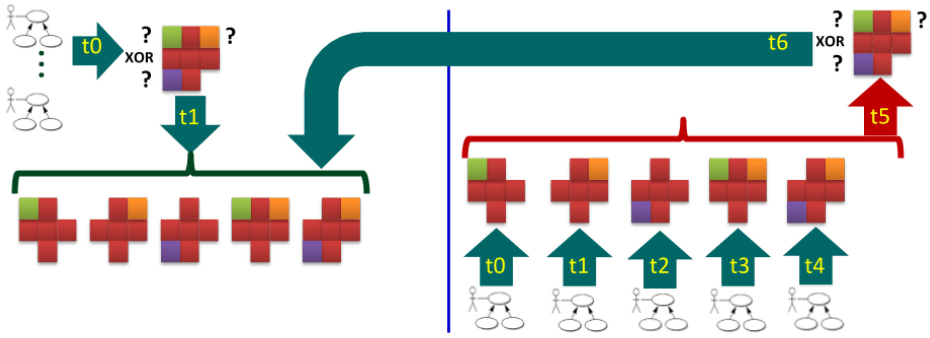


Fig. 1. Round-trip PLE adoption process

However, the industrial SotA in variability management is restricted to tools that automate top-down product variant generation from a variability model and reusable product assets, i.e. step t1 on the left of Fig. 1. No tool is currently available to automate the bottom-up extraction of a variability model and reusable PL assets, i.e. step t5 on the right of Fig. 1.

Companies thus face the software PL adoption dilemma: on the one hand, they are aware that PL can enable them to achieve large-scale productivity gains, improve time-to-market and product quality. On the other hand, however, these same companies already have existing variants created using the clone-and-own approach. This dilemma makes them practically unable to adopt PL. One solution to deal with this issue is to use round-trip engineering approach for PL adoption that consists in migrating, automatically or semi-automatically, the existing variants into a PL.

To conclude, innovative companies thus face the PLE adoption dilemma: the Return on Investment (ROI) of the proactive PLE adoption process is too uncertain, while the cost of late manual PLE is prohibitive. This dilemma considerably hinders PLE adoption. Many organizations eschew it, missing out on the massive long-term cost, robustness, customization, and competitiveness benefits that it would bring about for maintaining and developing their product portfolio. The REVaMP² project aims to provide the first solution to this dilemma by developing and validating on diverse industrial case studies, the first comprehensive round-trip engineering automation platform and process to support extractive, bottom-up PLE adoption and maximize reuse of legacy assets.

3 REVaMP² Tool Chain

The REVaMP² project develops a number of tool sets for Round-Trip Product Line Engineering as shown in Fig. 2, including innovative tools and services for Legacy and PL Asset Visualization, PL Asset Extraction Automation, PL Asset Verification Automation and PL Asset Co-Evolution Automation.

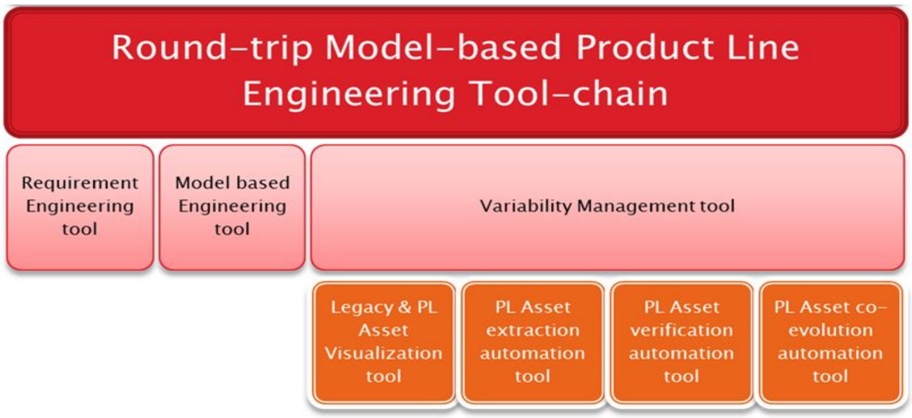


Fig. 2. REVaMP² Tool Sets for Round-Trip Product Line Engineering.

The first and second classes address the need to automate the extraction and visualization of product lines from legacy assets. This is needed because the extraction, verification and refactoring tools will not simultaneously reach 100% automation and quality. Human expertise will always be needed to adjust their parameters to trade-off automation for quality, evaluate their results and manually edit them. The realistic goal of REVaMP² is to minimize such manual edition steps, not to entirely eliminate them. The third class addresses the need to automate the formal verification of constraints on product line variability models and assets. These constraints can be for example, inter-feature consistency constraints, safety and real-time constraints that must hold for the whole configuration space or the existence of a nonempty intersection of this space with some business configuration goal. The fourth class addresses the need for PL refactoring automation. Next paragraphs summarize the current stage of the tools and services related to extraction and visualization and verification.

PL Asset Extraction and Visualization Automation. The tool sets related to extraction and visualization take as inputs the legacy assets as illustrated in Figure 3. Input legacy assets refer to any artefact needed to create a product and which are implemented without an explicit management of variability. For instance, systems that are implemented using the clone-and-own ad hoc reuse technique. The objective of the extraction and visualization tool sets is to analyse these legacy assets to extract the common and variable parts. The extraction process provide as output an explicit description of the variability in what is referred to as variability model eirshed with constraints that describe dependencies between variations points. It can also refactor the input asset to create reusable assets. Many challenges are identified in the context including the need to analyse and compare legacy assets. In addition, the extraction tools should support a variety of assets types ranging from textual requirements to the source code assets (in many different languages). Another identified challenge is to propose solutions to help and assist domain experts in the extraction process. REVaMP² aims implementing a tool chain including different tools to support the different asset types and including visualization supports to assist domain experts in this process (cf. Fig. 3).

At the current stage, many tools are implemented by the REVaMP² partners. This includes the following tool sets implemented by academics as well as industrial companies participating to the REVaMP² project: *BUT4Reuse* [9] framework from partner Sorbonne University, *VEXA* from partner ForschungsZentrum Informatik - FZI, *KernalHaven* [10] from partner University of Hildesheim, *Jittac Feature Filter* by Karlstad University [11], Tom Sawyer Visualization from partner Scopeset [12], *FLiMEA* from partner University San Jorge, *pure::variants variability framework* from partner pure::systems [13], *M-XRAY Architectural* analysis from partner MES [14]. In addition, to the variety and richness of the implemented tools, special attention is now devoted to the integration aspect where the objective is to create a tool chain including all the individual tools.

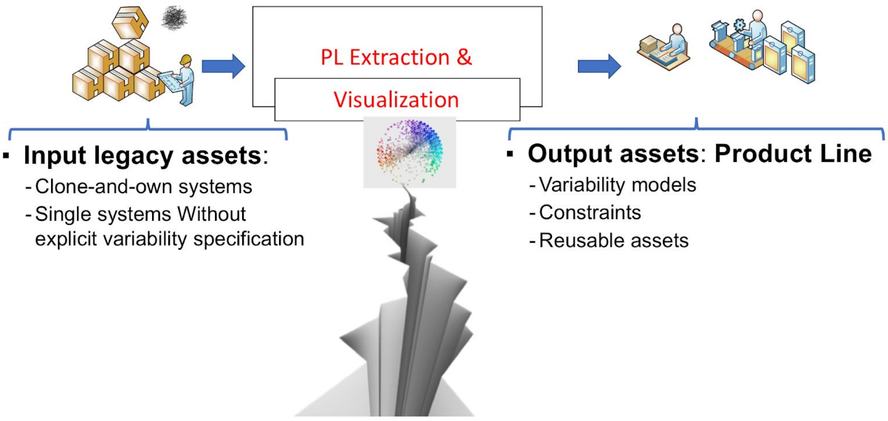


Fig. 3. PL Asset Extraction and Visualization Automation

The **PL Asset Verification** team works on developing tools assisting the PL engineering team verifying various kinds of PL artefacts using a variety of techniques. The current tool set includes the following tools: *Verification Studio* from partner Knowledge Centric Solutions, The Reuse Company [15], *AssetVerifier* from partner Kungliga Tekniska Högskolan - KTH, *KernelHaven* from partner Stiftung Universität

Hildesheim - SUH [10], *DragonflyME* from partner ForschungsZentrum Informatik - FZI and *VariaMos* from partner Université Paris 1 Panthéon-Sorbonne - UP1PS [16].

Verification Studio supports the verification of the individual correctness, global consistency and completeness of requirement artefacts. It is part of KCS-TRC's *Systems Engineering Suite (SES)* that also includes complementary tools allowing the engineering team to specify an ontology of the PL domain model and associate with each concept and relation of the ontology a set of natural language templates, each one corresponding to a way to express it in a textual requirement specification. SES also includes a requirement editor that leverages these templates to auto-complete requirement specification sentence fragments thus insuring that the requirement text only contains phrases which semantics is defined in the ontology. Verification Studio provides as built-in the requirements quality metrics defined by the INCOSE Guide for Writing Requirements.

AssetVerifier includes an editor for the formal specification in first-order logic of individual requirements of an automotive system PL together with their dependencies, variability model and required *Automotive Safety Integrity Level (ASIL)*. AssetVerifier relies on a *Satisfiability Modulo Theory (SMT)* solver to scalably verify for given target PL configuration (a) the consistency of the requirement dependencies an (b) that the ASIL are assigned in accordance with the rules mandated by the automotive industry safety standard ISO26262. AssetVerifier also includes an editor to annotate C code functions with pre and post-conditions constraints in the same formal language used for the requirements specification. It allows AssetVerifier to reuse its SMT solver to verify that the annotated C code satisfies the corresponding requirements.

DragonflyME supports modeling using a UML profile a virtual prototype PL of a real-time embedded system PL. The variability model of the PL is imported from an external tool such as pure::variants from pure-systems. For a given PL configuration, DragonflyME can generate the structural code of a virtual prototype allowing to run performance tests of the configuration.

KernelHaven supports the incremental computation of a great variety of PL quality metrics after each commit which affects the feature model and variable assets of the PL. It also allows the verification of the consistency between an abstract feature model and its operationalization in C code by #ifdef statements in C pre-processor files. It relies on a SAT solver to perform this verification task.

VariaMos supports the computation of quality metrics defined over variability models following an arbitrary meta-model. It also supports the detection of feature model defects such as dead features, redundant features, false optional feature and false and void feature models. For this task, it relies on a finite domain constraint solver.

The overall approach is supported modeling tools such as Modelio, requirements tools including REUSE tool set, and commercial product line engineering tool - pure::variants.

4 Results of the First Evaluation Phase and Outstanding challenges.

The REVaMP² project has provided the first set of results that were evaluated by the industrial partners providing the Use Cases (UCs). The primary goal for this initial evaluation was to depict the relationships between the different Use Case providers and Technology providers to enhance the SIS PL methods and tools as we know them today and to identify the gaps.

One of the main advantages that we could identify from the beginning of the project is the variety of industrial contributors providing the needs of different industries such as Aerospace, Automotive, Electronics. Those needs are addressed by a number of service, technology providers from academia and industry. This variety provides additional value to the solution that is to be applicable to any interested organization outside the project.

The analysis performed during the project illustrates a solution addressing the most common needs identified by the industry. The key assets that have been considered to evaluate the framework status are two, firstly Use Case Software Demonstrators, in which industry providers showcase the industrial challenges for PLE and possible solutions implemented with the help of one or several REVaMP² technology providers.

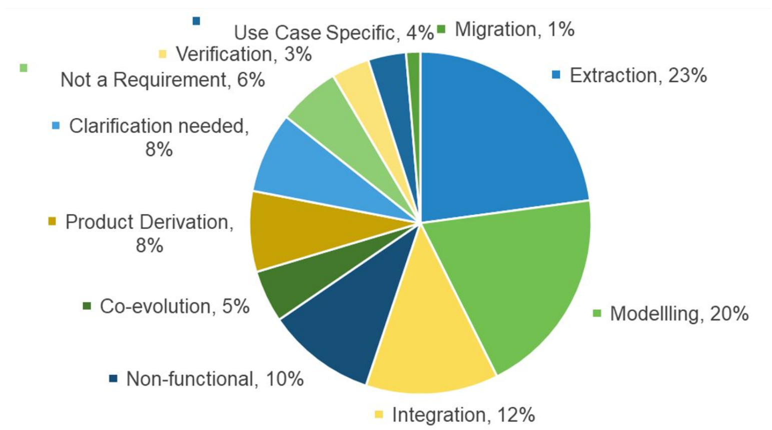
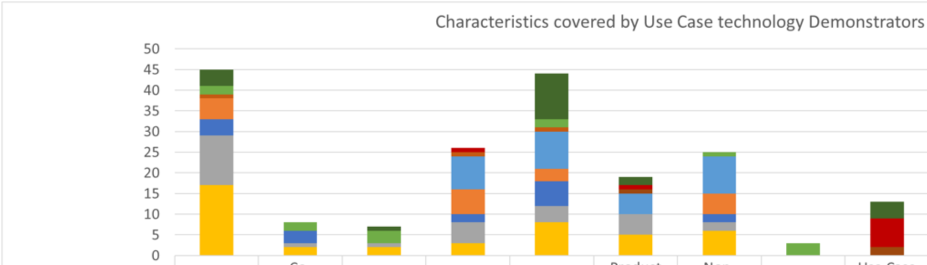


Fig. 4. Distribution of the main type of requirements from Use Cases satisfied by the Technology Demonstrators.

Use Case providers categorised their requirements into different typologies, so that the requirements could be mapped to a related technology to address the PLE challenges. We have identified more than 150 requirements from the first evaluation of the use cases, which has been continuously evolving to ensure the feasibility of the needs established at first in each of the UC. In Fig. 4. the distribution of these requirements among the most relevant typologies is illustrated.

The abovementioned distribution of requirements among the different types is the starting point for the allocation of requirements from the UC into the different capabilities covered by the technology providers. The results of this analysis is illustrated in Fig 5.



	Extraction	Co-evolution	Verification	Integration	Modelling	Product Derivation	Non-functional	Migration	Use Case Specific
IT Industry	4	0	1	0	11	2	0	0	4
Aerospace	2	2	3	0	2	0	1	3	0
Automotive	0	0	0	1	0	1	0	0	7
IT Industry									
Automotive	0	0	0	0	0	1	0	0	2
Trucks	1	0	0	1	1	0	0	0	0
Defense	0	0	0	8	9	5	9	0	0
IoT	5	0	0	6	3	0	5	0	0
Road Traffic Control	4	3	0	2	6	0	2	0	0
Automotive	12	1	1	5	4	5	2	0	0
Automation Industry	17	2	2	3	8	5	6	0	0

Fig. 5. Distribution of the main characteristics covered by Use Case technology demonstrators.

The requirements distribution clearly indicates the focus on challenges in extraction of the PL. In addition, the modelling is second large category. The co-evolution, one of the axes in REVaMP^2 project, was not highlighted by the requirements. However, it is a global understanding that co-evolution, i.e. maintenance of the product line over time is very important.

As a result of the integration of technologies to satisfy the different industry need, partners developed several UC Demonstrators based on REVaMP^2 tool chain. Fig 5. depicts a subset of tools used in several Use Cases in the first half of the project. The tools such as Eclipse Capra, FLIMEA and Jittac are analysed to be used in the following stages of the project.

USE CASE provider → TECHNOLOGY provider ↓	Automation Industry	Automotive	Road Traffic Control	IoT	Trucks	IT Industry
BUT4Reuse (Sorbonne University)	Extraction		Extraction			Extraction
C Code Verifier (Scania and KTH)			Modelling and/or Visualiz		Modelling and/or Visualization	
Crystal Bulb (Magillem)				Extraction / Verification		
Eclipse Capra (Univ. of Gothenburg)	TBD		TBD	TBD		
FeDeV-TS (ScopeSET)	Modelling and/or Visualiz					
FLiMEA (USJ)	TBD					
FINALIsT2 (ABB)	Extraction					
Jittac (Karlstad University)	TBD					
KernelHaven (SUH)	Verification	Extraction				
M-XRAY (MES)						
pure::variants DIMACS Exporter (Bosch)		Modelling and/or Visualization				
ReVaMP2 Plugin (TRC)			Verification			Extraction / Verification
VEXA (FZI)	Extraction	TBD				

Fig. 5. Matrix on what technology providers cover for the different industry needs.

As it is indicated above, on overall, the Use Cases confirm the initial assumption on the need for automation in a bottom-up PLE. The first evaluation results show interest in industry for extraction, modelling and verification of PLs. The major challenge is the need for integration of various tools for specific toolchains dedicated to Use Cases.

5 Conclusions

REVaMP[2] has already delivered a number of artifacts that are in active use by the partners within the project and outside of it. Importantly, many of the industrial tool providers have already integrated concepts and technology developed within the project into their offerings. Model Engineering Solutions MX-RAY [14] has, e.g., been extended to automatically extract architectural assets from the analysed models. Likewise, Siemens Industry Software has increased the technology readiness level of the product line support in LMS Imagine.Lab [17] for mechatronic system simulation.

The partners within the project are also working on new offerings for their customers or for internal use. ScopeSet is working on providing state-of-the-art feature and feature dependency visualisation capabilities based on technology developed in REVaMP[2]. Automotive and Industrial control partners have developed specialised internal tools during the project that support engineers with constructing safety arguments for a product line and with feature location in C/C++ codebases respectively.

Furthermore, work has been conducted on several open source projects that provide reverse engineering capabilities or supporting functionality. One notable example is *BUT4Reuse* [9] which provides commonality and variability analysis, feature identification, feature location, feature constraints discovery, feature model synthesis and other functionality. *KernelHaven* [10] is a powerful tool suite for analysing product lines that, among many other things, can identify unused code and configuration mismatches. *VariaMos* [16] supports its users in the modeling of product lines and the analysis of these models. *Eclipse Capra* [18] supports traceability between the assets of a product line and thus ties feature, source code, models, and test together, thus enabling change impact analysis and improved program comprehension. *Revamp2Plug-in* [15] provides wide functionalities from identifying variability and commonality in requirements to measuring Consistency and completeness quality of the assets involved in the product.

REVaMP^2 has also produced a number of notable project deliverables [19], for instance an overview of the state of the art of practices and tools for product line reengineering. Of course, the project partners are also very active in the scientific community. With more than 50 publications, the project has had a significant impact on the state of the art, with notable publications at ASE [20] , Isola [21], MODELS [22] as well as in IST [22], [23], TSE [24] and many others. Members of the project have also organised the main scientific event of the product line engineering community, SPLC in Gothenburg in 2018, with well over 100 participants and workshops and tutorials geared directly towards the topics of the projects.

Finally, the REVaMP^2 partners pure-systems, Thales, KTH, and Siemens are driving the standardisation of the *Variability Exchange Language* (VEL) in the context of OASIS [25]. They are joined by Dassault Systems, Intel, Accenture and PTC in the preparation of a standardised way to exchange variability information between different tools. This illustrates the relevance and impact of the results of REVaMP^2 beyond the project consortium and serves as an example of how the project results are disseminated to other interested parties.

Acknowledgement. This work was partially supported by the ITEA3 15010 REVaMP^2 project: FUI the Ile-de-France region and BPI in France, by Vinnova Sweden, and CDTI in Spain.

References

[1] A. Sadovykh, A. Bagnato, J. Robin, A. Viehl, T. Ziadi, and J. Martinez, “REVAMP: Challenges and innovation roadmap for variability management in round-trip engineering of software-intensive systems,” *Revue Genie Logiciel*, vol. 120, pp. 32–36, May 2017.

[2] J. Martinez, T. Ziadi, T. F. Bissyandé, J. Klein, and Y. Le Traon, “Bottom-up adoption of software product lines,” *Proceedings of the 19th International Conference on Software Product Line - SPLC ’15*. 2015.

[3] S. Apel, D. Batory, C. Kästner, and G. Saake, *Feature-Oriented Software Product Lines: Concepts and Implementation*. Springer Science & Business Media, 2013.

[4] C. Krueger, “Easing the Transition to Software Mass Customization,” *Software Product-Family Engineering*. pp. 282–293, 2002.

[5] F. van der Linden, *Software Product-Family Engineering: 4th International Workshop, PFE 2001 Bilbao, Spain, October 3-5, 2001 Revised Papers*. Springer, 2003.
[6] C. Kastner, A. Dreiling, and K. Ostermann, “Variability Mining: Consistent Semi-automatic Detection of Product-Line Features,” *IEEE Transactions on Software Engineering*, vol. 40, no. 1. pp. 67–82, 2014.
[7] T. Berger *et al.*, “A survey of variability modeling in industrial practice,” *Proceedings of the Seventh International Workshop on Variability Modelling of Software-intensive Systems - VaMoS ’13*. 2013.
[8] Y. Dubinsky, J. Rubin, T. Berger, S. Duszynski, M. Becker, and K. Czarnecki, “An Exploratory Study of Cloning in Industrial Software Product Lines,” *2013 17th European Conference on Software Maintenance and Reengineering*. 2013.
[9] “BUT4Reuse.” [Online]. Available: https://but4reuse.github.io/. [Accessed: 26-Jun-2019].
[10] KernelHaven, “KernelHaven,” *GitHub*. [Online]. Available: https://github.com/KernelHaven/KernelHaven. [Accessed: 26-Jun-2019].
[11] J. Buckley, S. Mooney, J. Rosik, and N. Ali, “JITTAC: A Just-in-Time tool for architectural consistency,” *2013 35th International Conference on Software Engineering (ICSE)*. 2013.
[12] “Tom Sawyer Visualization | Tom Sawyer Software.” [Online]. Available: https://www.tomsawyer.com/products/visualization/. [Accessed: 26-Jun-2019].
[13] “pure-systems - product line and variant management tools.” [Online]. Available: https://www.pure-systems.com/products/pure-variants-9.html. [Accessed: 26-Jun-2019].
[14] “MES M-XRAY: Consistent Metrics of Models - MES.” [Online]. Available: https://model-engineers.com/en/quality-tools/mxray/. [Accessed: 26-Jun-2019].
[15] “The REUSE company.” [Online]. Available: https://www.reusecompany.com/.
[16] SPLA, “SPLA/VARIAMOS,” *GitHub*. [Online]. Available: https://github.com/SPLA/VARIAMOS. [Accessed: 26-Jun-2019].
[17] “Simcenter System Simulation,” *Siemens PLM Software*. [Online]. Available: https://www.plm.automation.siemens.com/global/fr/products/simcenter/simcenter-system-simulation.html. [Accessed: 26-Jun-2019].
[18] S. Swart, “Eclipse Capra,” *projects.eclipse.org*, 28-Jul-2016. [Online]. Available: https://projects.eclipse.org/projects/modeling.capra. [Accessed: 26-Jun-2019].
[19] “REVAMP2 Projects Public Deliverables.” [Online]. Available: http://www.revamp2-project.eu/publications/public-project-results.
[20] Mukelabai, M. Mukelabai, D. Nešić, S. Maro, T. Berger, and J.-P. Steghöfer, “Tackling combinatorial explosion: a study of industrial needs and practices for analyzing highly configurable systems,” *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering - ASE 2018*. 2018.
[21] M. Nyberg, D. Gurov, C. Lidström, A. Rasmusson, and J. Westman, “Formal Verification in Automotive Industry: Enablers and Obstacles,” *Lecture Notes in Computer Science*. pp. 139–158, 2018.
[22] M. Ballarín, A. C. Marcén, V. Pelechano, and C. Cetina, “Measures to report the Location Problem of Model Fragment Location,” *Proceedings of the 21th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems - MODELS ’18*. 2018.
[23] S. El-Sharkawy, N. Yamagishi-Eichler, and K. Schmid, “Metrics for analyzing variability and its implementation in software product lines: A systematic literature review,” *Information and Software Technology*, vol. 106. pp. 1–30, 2019.
[24] L. Passos *et al.*, “A Study of Feature Scattering in the Linux Kernel,” *IEEE Transactions on Software Engineering*. pp. 1–1, 2018.
[25] “OASIS Variability Exchange Language (VEL) TC | OASIS.” [Online]. Available: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=vel. [Accessed: 26-Jun-2019].