Enhancing the Feature Retrieval Process with Scoping and Tool Support – PAxSPL_v2

Luciano Marchezan[1]. João Carbonell[1], Elder Rodrigues[1], Maicon Bernardino[1], Fábio Paulo Basso[1] and Wesley K. G Assunção[2].

Third Workshop on Experiences and Empirical Studies on Software Reuse

lucianomarchp@gmail.com

[1]Federal University of Pampa (Unipampa)
[1]Federal University of Technology - Toledo PR

Overview

Introduction

- The benefits and motivation for adopting SPL include reduced maintenance effort, reduced time-to-market, increased reusability [PBvDL05].
- The extractive adoption of SPLs is performed by the conduction of a reengineering process [ALHL+17].
- Due to the importance of the reengineering process, in previous work, we proposed PAxSPL [MMRBPB19]

Motivation

- After the execution of a case study for evaluating PAxSPL, we identified the need for improvements for better adherence to industrial contexts.
- The limitations led us to evolve PAxSPL into a new version, PAxSPL_v2.

Contributions

The contributions of our work are:

1. Evolution of the PAxSPL to include SPL scoping activities, allowing SPL scoping customization for specific contexts.
2. Development of a supporting tool that can be used for executing all PAxSPL activities.
3. Empirical results of a pilot study to collect evidence about how PAxSPL may be instantiated to a real scenario, and to identify possible limitations.

Limitations Identified in PAxSPL_v1

- The lack of SPL scoping related activities when conducting the SPL reengineering.
- High effort of some activities when compared to others.
- The need of keeping artifacts generated during the reengineering updated, organized, and well-structured.

PAxSPL_v2 Process

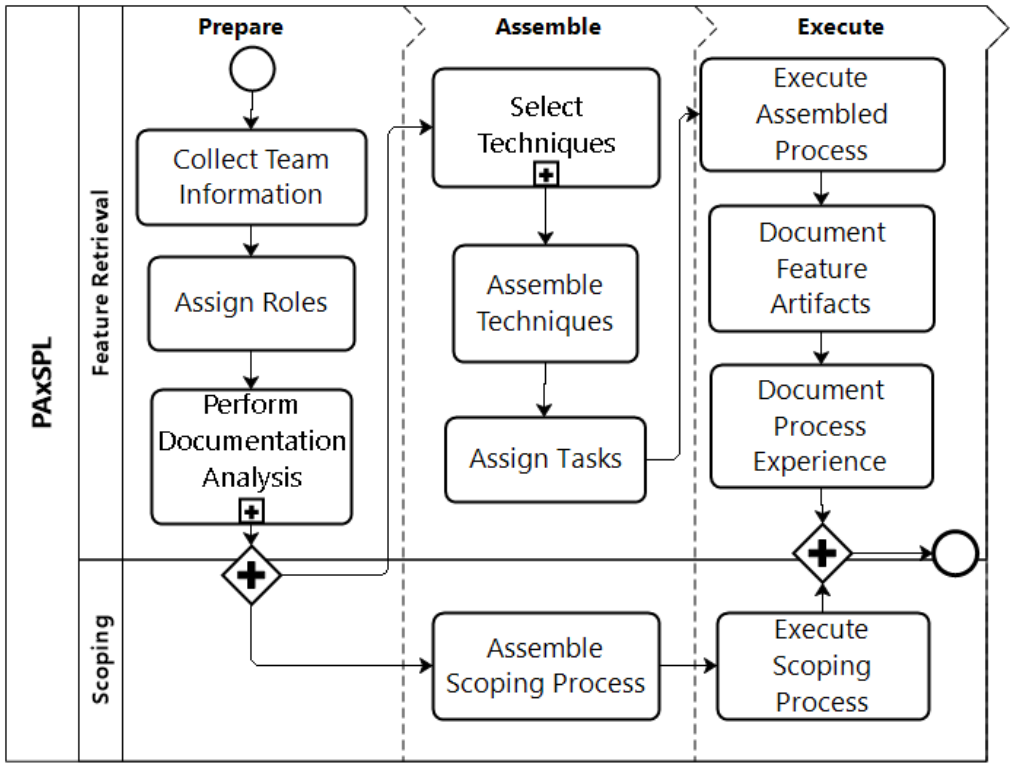
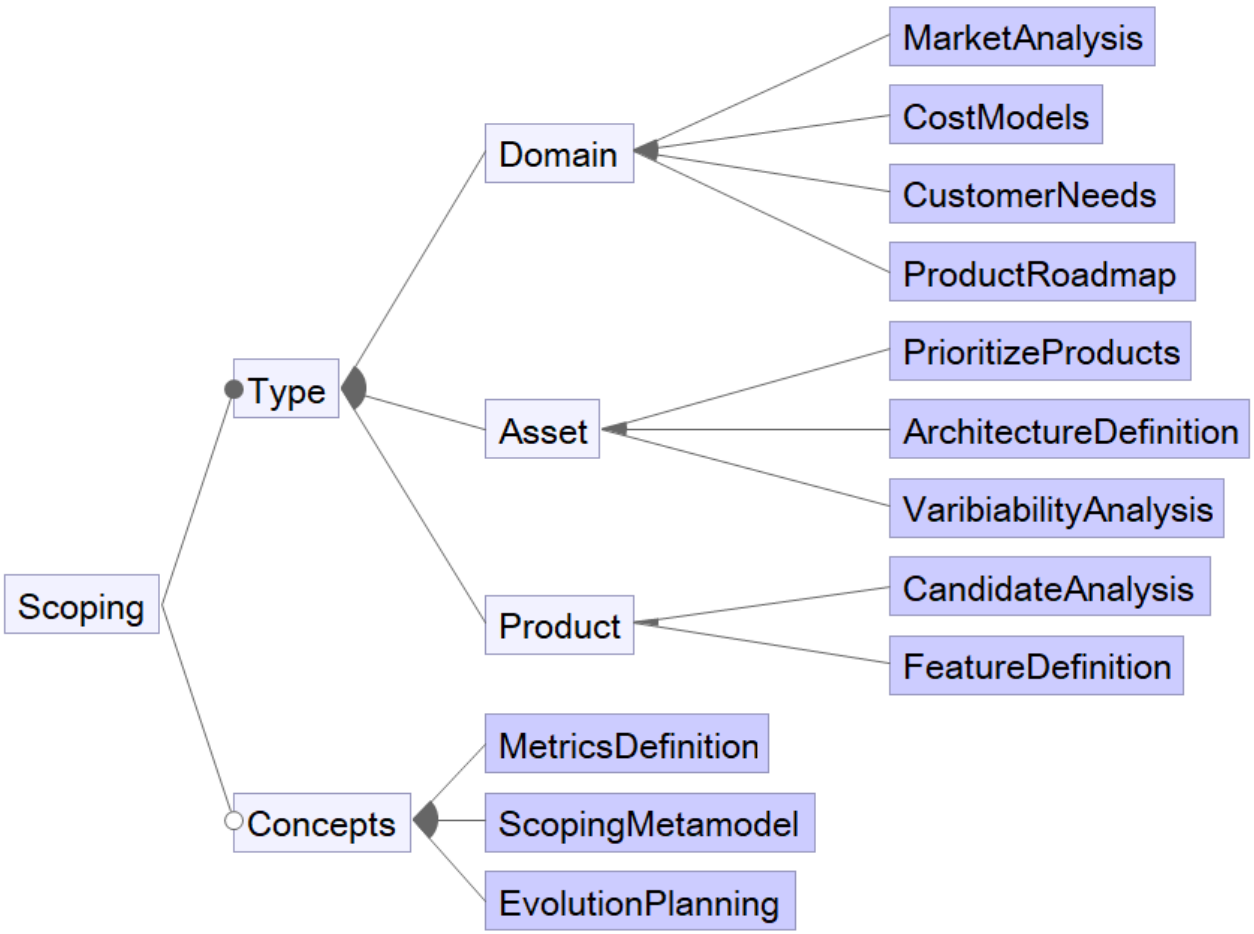


Figure 1: PAxSPL_v2 including SPL Scoping Activities

PAxSPL_v2 Scoping Support



Supporting Tool

- The tool supports the execution of all activities of the process life cycle.
- Aims at reducing the effort when conducting the feature retrieval and scoping process.
- An important part of the tool is feature analysis. Users can register the features retrieved, their descriptions, relations, and types.

Figure 2: Tool is available at http://paxspl-tool.herokuapp.com/

Supporting Tool

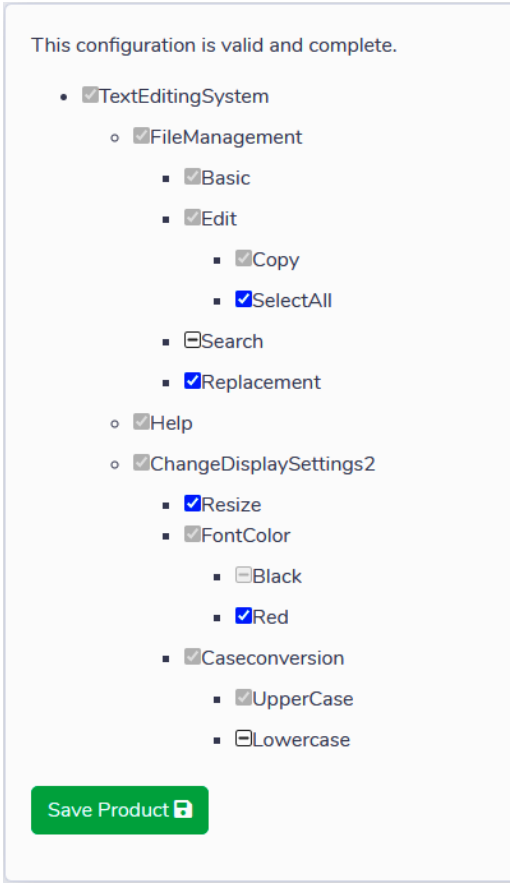


Figure 3: Product Configurator Screen

Pilot Study

The main goal of this evaluation is to measure PAxSPL adherence and expressiveness in terms of customization.

1. **RQ1.** Can PAxSPL_v2 be instantiated to represent the activities, artifacts, and techniques used in scenarios were SPL reengineering was conducted?
2. **RQ2.** What limitations are observed by customizing PAxSPL_v2?

Data Set

We selected studies from the ESPLA catalog [MAaZ17] by applying three criteria to them:

1. The study must apply at least one retrieval technique present in the PAxSPL_v2 guidelines;
2. The study presents a scenario different from other approaches already selected;
3. The study evaluation protocol, data set and results are available online.

Procedure

After selecting the study, we performed these steps:

1. Identify and register inputs and output artifacts;
2. Identify and register feature retrieval techniques used;
3. Identify the feature retrieval activities and their workflow;
4. Use in PAxSPL_v2, with the tool supporting, the artifacts, techniques, and activities identified.

Procedure

After executing these steps, we plan to answer the RQs by analyzing:

- The number of artifacts from the original study that were used by PAxSPL_v2 (**RQ1**);
- The retrieval techniques that were assembled into PAxSPL generic process (**RQ1**);
- The activities that were assembled into our framework (**RQ1**);
- The limitations faced during execution of the study (**RQ2**).

Execution

Table 1: Data from the Original Study

Ref.	Artifacts	Tech.	Activities
[ESSD13]	object-oriented source code; feature descriptions	LSI; FCA; Clustering;	(i) Use LSI to divide features and classes into common and variable partitions; (ii) fragment variable partitions into minimal disjoint sets using FCA; (iii) derive code-topics from common class partition; (iv) perform traceability links between features and their code-topics; (v) determine which classes implement each feature.

Results

- We were able to assemble and execute their process into the PAxSPL tool.
- Five activities and their workflow were assembled based on the original study, however, minor modifications were made, such as writing the first words as verbs.

Results

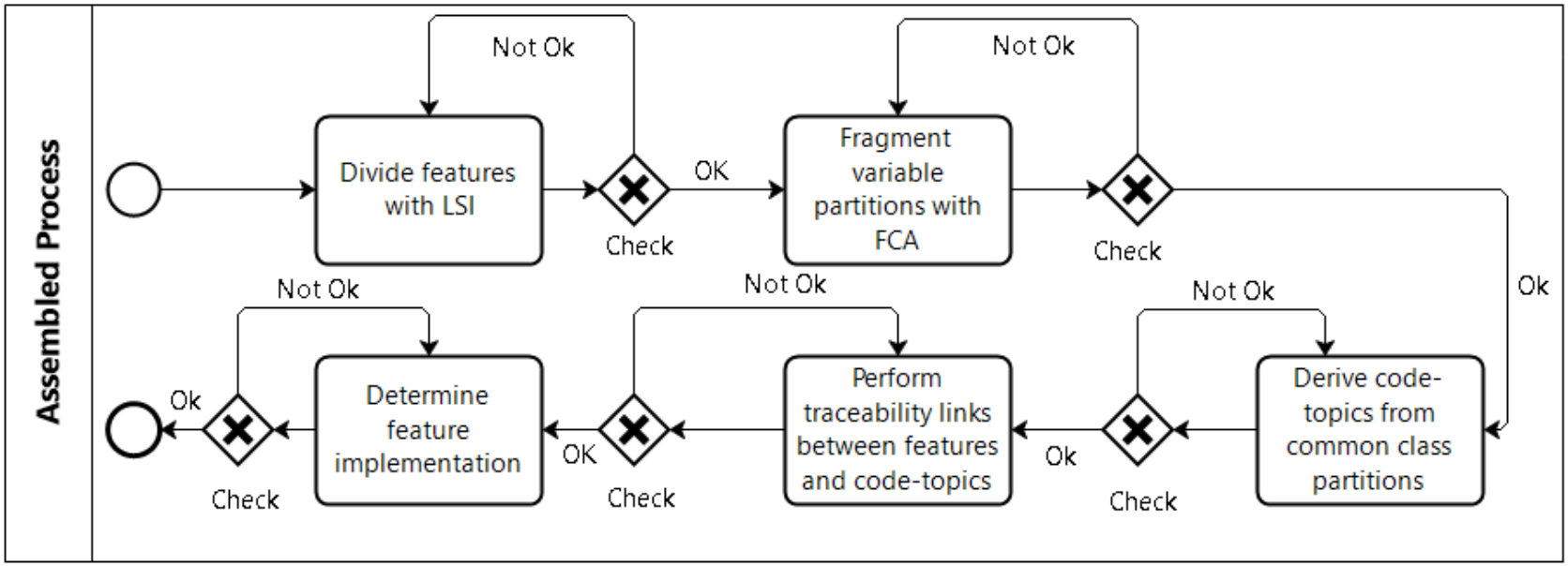


Figure 4: BPMN Process Assembled for [ESSD13]

Results

As this is a pilot study, we only have preliminary results for answering our RQs.

- **RQ1**: We were able to include all input artifacts and use them during the execution.
- This may indicate that PAxSPL_v2 was accordingly instantiated to the scenario.
- However, we should include new scenarios to collect further evidence to support this claim.

Results

As this is a pilot study, we only have preliminary results for answering our RQs.

- **RQ2**: We identified that the generic feature retrieval process may have to be changed.
- Before performing changes in the generic process, however, we also intend to perform the evaluation with other scenarios.

Conclusion

- By including SPL scoping concepts and specific activities, we extended PAxSPL's life-cycle.
- We developed a web-based supporting tool for conducting PAxSPL activities.
- With the goal to study PAxSPL_v2 instantiation capabilities, we defined a protocol for empirically evaluating our framework.
- In this pilot study, we were able to apply our framework in a scenario.
- We plan to extend the results of the evaluation by including more scenarios extracted from the catalog.

References I

W. Assunção, R. Lopez-Herrejon, L. Linsbauer, S. Vergilio, and A. Egyed, *Reengineering legacy applications into software product lines: a systematic mapping*, Empirical Software Engineering **22** (2017), no. 6, 1–45.

H. Eyal-Salman, D. Seriai, and C. Dony, *Feature-to-code traceability in a collection of software variants: Combining formal concept analysis and information retrieval*, 14th International Conference on Information Reuse and Integration, IEEE, 2013, pp. 209–216.

References II


Jabier Martinez, Wesley K. G. Assunção, and Tewfik Ziadi, *Espla: A catalog of extractive spl adoption case studies*, Proceedings of the 21st International Systems and Software Product Line Conference - Volume B (New York, NY, USA), SPLC '17, ACM, 2017, pp. 38–41.


Luciano Marchezan, Elder Macedo Rodrigues, Maicon Bernardino, and Fábio Paulo Basso, *Paxspl: A feature retrieval process for software product line reengineering*, Software: Practice and Experience **49** (2019), no. 8, 1278–1306.


K. Pohl, G. Böckle, and F. van Der Linden, *Software product line engineering: foundations, principles and techniques*, Springer Science & Business Media, 2005.

Enhancing the Feature Retrieval Process with Scoping and Tool Support – PAxSPL_v2

Luciano Marchezan[1]. João Carbonell[1], Elder Rodrigues[1], Maicon Bernardino[1], Fábio Paulo Basso[1] and Wesley K. G Assunção[2].

Third Workshop on Experiences and Empirical Studies on Software Reuse

lucianomarchp@gmail.com

[1]Federal University of Pampa (Unipampa)
[1]Federal University of Technology - Toledo PR