
Karen D. R. Pacini
and Rosana T. V. Braga

Institute of Mathematics and Computer Sciences
University of São Paulo
São Carlos, SP, Brazil
Email: karenr@icmc.usp.br, rtvb@icmc.usp.br

Abstract—In order to successfully design and build a Software Product Line (SPL), besides the difficult task of making a good domain engineering based on a solid knowledge – both theoretical and practical – about the subject domain, it is still necessary to consider other barriers such as lack of computational support, lack of documentation available and the complexity or unavailability of existing supporting tools. These are some of the reasons that may discourage the adoption and wide usage of SPL in organizations. In this context, this paper presents a Systematic Mapping (SM) of supporting tools for managing SPLs. This SM was performed in order to identify, gather and classify existing solutions in the literature that offer support for managing product lines both in single or multiple phases, since conception until product derivation and evolution of the SPL. The information gathered about the solutions selected is presented in the results. This information comprises the completeness of the solutions, their complexity and quality, and also points out their benefits and limitations. It is expected as the result of this SM an overview of SPL management solutions in order to support developers and SPL engineers to find suitable options to apply in their projects, in addition to highlight gaps on the research area and suggest future works.

Keywords—Software Product Line Management; Systematic Mapping.

I. INTRODUCTION

The software industry has been adapting to the large increase of demand arising from the constant evolution of technology. The concept of software reuse gets an important role on this new way of software manufacturing, in which development time is reduced, while quality is improved [1]. Over time, many approaches have emerged trying to achieve this goal, such as: object-oriented paradigm, component-based development, service-oriented architecture, among others.

Software product lines (SPL) emerged in this context, to support reuse by building systems tailored specifically for the needs of particular customers or groups of customers. Reuse in SPL is systematic, it is planned and executed for each artifact resulting from the development process. According to Ezran et al. [2] these artifacts encapsulate knowledge and are very valuable to the organizations, because they are an interrelated collection of software products that can be reused across applications.

The most common SPL development approaches, such as Product Line UML-Based Software Engineering (PLUS) [3], Product Line Practice (PLP) [1], etc., are focused on the process to support the domain engineering and/or the application engineering, without considering the computational tools that support the process. Thus, the choice and use of tools are apart from the process and strongly associated to phases of definition of the feature models and its mapping to the artifacts that implement them. Some examples of these tools include Pure::Variants [4], Gears [5], and GenArch [6].

Supporting tools offer the developers a complete environment for development and maintenance of the product line, aiming at facilitating its adoption. When supporting tools are employed for SPL management, from conception to evolution, developers can dedicate more attention to the development itself, i.e., to domain and application engineering. This can help improving production quality for both the product line and its generated products, deviating the focus from the development environment or other more specific management questions. Although there is a huge variety of existing tools, it is not possible to ensure that the needs of SPL engineers are being fulfilled. It is necessary to better investigate the scope, the availability and the specificity of these supporting tools in order to identify gaps to be filled.

Therefore, this paper presents a Systematic Mapping (SM) of supporting tools for managing SPLs in order to identify and analyse the solutions that exist in the literature. The identified solutions can support the management of SPL from conception to development, maintenance and evolution phases. The analysis proposed in this paper intends to offer an updated overview of the existing supporting tools and identify perspectives of researches related to product lines.

Section II presents some relevant related works that resemble this work in terms of searching for tools that support somehow the development of SPL. Section III presents the process of the SM, with the corresponding phases: planning, conducting and reporting. Section IV presents the summarization and the data analysis observed from the studies selected from the SM. Section V presents a discussion about the analysis of the gathered solutions, pointing out its benefits and limitations. Finally, Section VI presents the conclusions of this research.

II. RELATED WORKS

SPLs have become a popular concept nowadays. Despite the difficulty on its adoption, there are big investments and research improvements in this area. Therefore, there are many
This SM followed the same process proposed by Kitchenham for SLRs [8], which contains three main phases: 1-Planning: the research objectives and the SM protocol are defined; 2-Conducting: the primary studies are identified, selected, and evaluated according to the inclusion and exclusion criteria previously established; and 3-Reporting: a final report is organized and presented according to the data extraction and analysis.

A tool for supporting the SM process called StArt (State of the Art through Systematic Review) [15] was used to manage the whole execution of this SM. Figure 1 illustrates how StArt deals with the several SM phases. The full protocol and SM StArt File with the full process, all outcomes (including the list of the 1046 works, where the 50 selected ones appear in the respective Extraction phase), filled forms and other details can be found elsewhere for further reference [16]. This extra material allows the reproduction of the study whenever necessary, e.g., to update it or to evolve it to a SLR.

The next subsections present the three main phases of the SM process in detail.

A. Planning

In this phase, the SM protocol is established, which consists of: research objectives; research questions, range, and specificity; sources selection criteria; studies definition; and procedures for studies selection.

1) Objective: This SM aims to investigate the state of the art with respect to computational support to the SPL management, regarding the phases of development, maintenance and evolution of SPL, in order to identify the quantity and the quality of the solutions currently available, considering its completeness and complexity, as well as highlighting its benefits and limitations.

2) Research Question: Aiming at finding possibly all primary studies to understand and summarize evidences about existing solutions for SPLs management, the following research question (RQ) was established:

RQ1: What are the existing solutions in the literature that present a computational support to the management of SPL?
a) Does the presented solution use patterns or known standards?  
b) What is the technology used for SPL management presented by the solution?  
c) Which phases of the SPL life cycle are supported by the solution?

By management we mean inclusion, modification, removal, or search of all the artifacts produced during the SPL engineering (domain engineering or application engineering), as well as its evolution after the SPL is delivered.

3) Inclusion and Exclusion Criteria: The Inclusion Criteria (IC) and the Exclusion Criteria (EC) make it possible to include primary studies that are relevant to answer the research questions and exclude studies that do not answer them. Thus, the inclusion criteria of this SM are:

- **IC1**: Studies that present a tool, approach, technique, process, method or any other software engineering resource that offers a solution for management of one or more phases of the SPL life cycle.
- **IC2**: Studies that present a proposal of a solution for managing one or more phases of the SPL life cycle, even if it is yet in the project phase.

Criteria IC2 reinforces that any practical solution that could effectively help the SPL management, even if the solution is still in project, is included, as it could give us insights about important issues to consider. The term practical, in this context, does not mean a concrete solution, but purely a solution that directly helps the SPL management.

Non-relevant studies with respect to the objectives of this SM are discarded applying one of these four defined exclusion criteria:

- **EC1**: Studies that do not present a solution, consolidated or not, for the management of one or more phases of the SPL life cycle.
- **EC2**: Studies that are short versions of a published full work.
- **EC3**: Studies that are incomplete, unavailable and/or duplicated (multiple instances of same document).
- **EC4**: Studies that describe events, or are an event index or schedule.

For studies classification, i.e., inclusion or exclusion, it is mandatory to apply one of the defined criteria above. However, if more then one criteria is applicable to a particular study, this formula is used: \(( (IC1 \text{ OR } IC2) \text{ AND NOT } ((EC1 \text{ OR } \text{EC2}) \text{ OR } (EC3) \text{ OR } (EC4)) \).

4) Sources Selection Criteria Definition: The sources choice was made considering their relevance in the software engineering area and also a specialist opinion in cases of conferences, books and workshops that are not indexed in the search engines. The specialist opinion included in several activities relied on the participation of the co-author of this paper. For the search engines, we considered those with an updated content, with availability of the studies, with an advanced search mechanism, with quality results and with flexibility to export the findings. Thus, the following search engines were selected: IEEE Xplore, ACM Digital Library, Science Direct, Scopus and Web of Science.

5) Search String Construction: From a group of studies selected by the specialist, called in the SM process the ‘control group’, in addition to the objective of this SM, the search string was defined according to Table I and resulted in: \((A) \text{ AND } (B) \text{ AND } (C) \text{ AND } (D)\).

<table>
<thead>
<tr>
<th>Term</th>
<th>Keyword</th>
<th>Synonym</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>Tool</td>
<td>&quot;tools&quot;, &quot;tool-supported&quot;, &quot;support&quot;, &quot;supported&quot;, &quot;supporting&quot;</td>
</tr>
<tr>
<td>C</td>
<td>Management</td>
<td>&quot;manage&quot;, &quot;managing&quot;, &quot;storage&quot;, &quot;repository&quot;</td>
</tr>
<tr>
<td>D</td>
<td>Software</td>
<td></td>
</tr>
</tbody>
</table>

B. Conducting

This phase was performed right after the protocol definition and was divided into two phases of selection. It was carried out between November, 2013 and January, 2014. The first phase is called studies identification, and it defines the group of studies that will be used as base to the second phase, which is called studies selection. In the first phase of our SM, the search was performed using five search engines (ACM Digital Library, IEEE Xplore, Science Direct, Scopus, and Web of Science), according to the previously defined string. The specific syntax was used in each search base considering only the title, the keywords and the abstract. This resulted in 1046 works.

In the second phase, we applied the inclusion and exclusion criteria defined in the SM research protocol. After the selection is performed, a refined group of studies is obtained according to the context of the SM for the data extraction.

The selection phase of this SM was divided into three parts. In the first selection the inclusion and the exclusion criteria were applied to the base group of identified studies considering only title and abstract, which resulted in the inclusion of 95 studies and the exclusion of 951 studies. Among the excluded studies, 366 were marked as duplicated and 585 were excluded by other exclusion criteria, resulting in 95 studies. In the second selection, the full text was considered, and 41 studies were included while 54 were excluded. The third selection used the specialist opinion in order to validate the selected studies and therefore, 9 previously excluded studies were included again by consensus. The Figure 2 and the Figure 3 present the total application of inclusion and exclusion criteria, respectively, in the selection of studies by search engine.

After the selection of the studies included in the SM, it is possible to proceed to the next phase of the process, where data extraction is performed.

C. Reporting

This phase aims to extract and analyze the data in order to organize and present a final report about the findings. The data extraction summarizes the data of the selected studies for further analysis. For the data extraction performed in this SM, an extraction form filled after reading each paper was used.

This form is intended to help answering the research questions of this SM. In addition to collecting the basic information about the studies, such as title, author, year and type of publication (journal, conference, etc.), the form also...
collects specific information useful for this research. Among the specific information, we can mention the type of solution presented by the study, a brief description of the solution, with its benefits and limitations, its specificity, the handled phases, the software engineering resources used in the solution, the use of patterns/standards, validation and managed parts.

IV. RESULT ANALYSIS

After a careful data extraction, sufficient information was gathered to perform an analysis of the results. The analysis of the selected studies was divided according to the characteristics described in the extraction form.

A. Full Analysis of the Studies

The analysis was divided into two parts, the first considers all solutions observed in the phase of data extraction, while the second part performs a more detailed review only on the most complete solutions.

1) Frequency of Publication: It was observed that the frequency of publications of solutions for the management of SPLs has increased significantly from 2008 to its peak in 2011, and then remained stable. This trend can be graphically observed in Table II, indicating that this is an area of current research interest.

<table>
<thead>
<tr>
<th>Year</th>
<th>2000</th>
<th>2001</th>
<th>2002</th>
<th>2003</th>
<th>2004</th>
<th>2005</th>
<th>2006</th>
<th>2007</th>
<th>2008</th>
<th>2009</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>9</td>
<td>5</td>
<td>6</td>
<td>9</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>IEEE</td>
<td>9</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Science Direct</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Scopus</td>
<td>17</td>
<td>13</td>
<td>3</td>
<td>17</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Total</td>
<td>44</td>
<td>16</td>
<td>3</td>
<td>17</td>
<td>9</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
</tr>
</tbody>
</table>

2) Specificity: A great part of the solutions found are specific to support only a particular cycle of SPL management, as can be observed in Table III. Among the specificity, we can highlight the variability management. In fact, most solutions available both commercially and freely are geared to support variability management, providing processes, models, methods, approaches, tools, among others. Examples are KobrA [17] and COVAMOF [18].

<table>
<thead>
<tr>
<th>Specificity</th>
<th>Architecture</th>
<th>Asset</th>
<th>Compatibility</th>
<th>Configuration/ Derivation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evolution</td>
<td>5</td>
<td>1</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>Requirement</td>
<td>18</td>
<td>1</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

3) Patterns/standard Use: Among the observed solutions the lack of pattern/standards used during the development becomes evident, as shown in Table IV. This may hinder the flexibility of the solution, and even its adoption and integration with other existing solutions. Although the use of patterns or standards may generate an additional effort, their use to develop a solution is strongly recommended, as the effort would be rewarded in the future with enhanced system maintenance and evolution.

<table>
<thead>
<tr>
<th>Patterns or Standards</th>
<th>RAS</th>
<th>OCL</th>
<th>CVL</th>
<th>ADL</th>
<th>CVA</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>IEEE</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>None</td>
</tr>
<tr>
<td>Science Direct</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Scopus</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td>40</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

The standards identified in the solutions presented in Table IV are: Reusable Asset Specification (RAS) [19], Object Constraint Language (OCL) [20], Common Variability Language (CVL) [21], Architecture description language (ADL) [22], Commonality and Variability Analysis (CVA) [23], Meta-Object Facility (MOF) [24], Orthogonal Variability Model (OVM) [25], Variability Language VML [26] and Query/View/Transformation (QVT) [27].

4) Phases Supported: From the 50 studies selected, only 17 offer support to both development phases (domain engineering and application engineering), in the remainder, five support only application engineering and 28 only domain engineering.

The information extracted from these 17 more relevant studies, which support both phases of development, are presented in detail in Table V, and are evaluated more specifically in the second phase of the analysis.
B. Selected Studies Analysis

Among the 17 solutions presented by the studies, 15 offer support to the development of the SPL, while two offer support only to maintenance and evolution of existing SPL. From the 15 solutions that support the development phase, one supports only this process and 14 offer support to maintenance too. Among these 14 solutions, three support only these two activities, one supports also the evolution of the SPL, one supports also the derivation of products and nine additionally offer a visualization of the SPL with its variability. Among these nine, only one also offers support for evolution.

Only two solutions offer support from the development to the maintenance and evolution of product lines, and only one of them offers an overview of the variability and the produced assets.

Thereby, the lack of solutions that support all phases of development and evolution of SPL is evident, which highlights a gap to be investigated.

V. DISCUSSION

The existing solutions in the literature for computational support to SPLs were identified in the SM performed in this work. In addition, it was possible to identify and analyse the major benefits and limitations of the overall solutions selected.

Among the benefits observed, we highlight that most of the solutions intend to provide guidance to resolve the problem or part of it, besides providing an execution flow to achieve a goal. These flows are presented as a process or even wizards, which allows users, both beginners and experts, to operate in a particular management area, such as SPL development or evolution, ensuring that the process was correctly executed and that its results are consistent, since the provided management is explicit and organized.

Besides that, seven solutions support the user when making decisions and understanding the process. Regarding the decision making support, six solutions provide the users both overviews and specific views from the current state of the SPL, highlighting the chosen objects to be observed.

Maintaining the consistency is also an issue treated by most solutions found (13 solutions), some establish constraints, inspections and validations, some even provide the tracking of the assets, which supports also the SPL evolution.

It is also worth to mention the great reduction of costs and effort provided by the nine solutions that offer various automated functionality to support the management. Five solutions even offer resources to significantly reduce the complexity of tasks that would be very laborious to perform manually.

The leverage of reuse is also an issue treated by 12 solutions. These solutions often provide configuration, importation and exportation options, compatibility with others tools and sharing of assets among product lines, which promotes the interoperability and leverage the reuse, both inside the solution itself and out of it.

However, despite the many benefits observed, it was possible to identify many limitations that often can discourage the use of the solution. Among the main limitations, the lack of complete solutions to manage SPLs is highlighted. Most of the analysed solutions (16 solutions) are focused on the resolution of a single issue, that is, provide support only to manage a specific area. Besides that, six solutions have a very marked limitation regarding the scope to address an area and can be very specific, for example, manage only requirements, and furthermore do not offer the possibility of extension or parametrization to generalize their use.

The authors of seven solutions declare that their use may be more complex than desired because, to use the solution, the user needs to study and master its specifications, which very often discourage its adoption. Besides the complexity, a barrier to be also considered is that most solutions analysed (10 solutions) need a great intervention of a specialist that understands the models and specifications and performs many manual procedures.

One of the limitations strongly considered as motivation for the adoption of a solution is the lack of a graphical interface, or an interface that is very complex or poor, as occurs in four solutions. One of the solutions does not even support version control, which makes the SPL maintenance and evolution very difficult.

A great barrier, mostly at the academy, is the fact that various solutions are private [37][40]. Unfortunately, those are the solutions that provide the most complete group of solutions for SPL management, as well as user support and validation. Some freely available solutions analysed have not been even validated or implemented yet.

VI. CONCLUSION AND FUTURE WORKS

This work aimed to identify and evaluate existing solutions in the literature to support the management of SPLs. For this, a SM was performed, in which the research protocol was defined and followed for conducting the research. From the data extracted from the selected studies, it was possible to gather enough information to answer the research questions proposed in this paper.

The outcomes of the research present 50 existing solutions in the literature that provide computational support for at least one phase of SPL management, which represents a very reduced offer of solutions to support the management of SPL. Most of the existing solutions do not offer a complete support, not covering the whole phases of the development, maintenance and evolution of the SPL and even not providing a great usage of the whole reuse potential that artifacts and features may offer.

The contribution of this paper is interesting both to the academic and to the professional segment. In the academic environment, this research helps to highlight the lack of complete solutions in this area, in addition to highlight the lack of standards on these existing solutions and the lack of validation, which hinders its use. In the professional context, this research helps professionals to find potential tools that will help them deploy a product line or even help them manage existing product lines.

The gaps pointed out in this SM suggest perspectives of future works, of which we can highlight:

- **Leverage of reuse:** Among all the presented solutions, only one [30] allows the sharing of the assets in the repository between SPLs. An approach where the stored assets may not be bounded into a specific SPL and could be available to be freely used in any context.
would leverage the reuse not just inside the SPL itself but widely.

- **Use of patterns/standards:** The lack of use of standards was evidenced in this research. A solution designed taking into account standards or patterns could provide more safety and organization for the user, besides promoting the flexibility and interoperability with other tools. As an example of the use of standards in solutions focused on SPLs, we highlight RAS [19] and CVL [21].

- **Use of services:** The service-oriented usage is also a good option in order to promote the interoperability among tools and facilitate the access to the functionality of the solution. This is a resource that was not found in most solutions analysed, except for the solution proposed by S. Khoshnevis [28].

- **A free complete solution to support SPL management:** The creation and provision of a free and complete approach to support the management of SPLs, with validation, documentation and adequate support could rise the community interest in adoption of the SPL concept, both for developing and usage.

- **Possibility of extending the SM towards a complete SLR:** The SM main goal was to identify all research related to a specific topic. Based on the resulting complete protocol that is publicly available [16], a complete SLR can be performed to address specific questions.

The solutions that can be built as a result of this SM could incentive the adoption of SPL in both academical and professional environments, in addition to leveraging the reuse of assets in contexts other than SPL. As the next step of this research, it is planned to analyse existing solutions, by running performance and quality tests in order to compare and evaluate them through a full systematic review about supporting tools for the management of SPLs. Then, if none of the solutions are considered adequate, we aim to propose a new solution that fulfills developers needs.

ACKNOWLEDGMENTS

Our thanks to Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) and University of São Paulo (USP) for financial support.

REFERENCES


