



A REVIEW OF SOFTWARE PRODUCT LINE ADOPTION

¹M.A JABAR, ²ZAREI M.B., ³SIDI F, ⁴N.F.M SANI

^{1,2}Department of Information System

^{3,4}Department of Computer Science

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

¹marzanah@upm.edu.my, ²bahareh_zarei_mn@yahoo.com, ³fatimah@upm.edu.my, ⁴fazlida@upm.edu.my

ABSTRACT

Software Product Line was proposed in 2000 as a systematic approach toward reuse. Although benefits of this strategy is totally well known but there are still uncertainties between organizations about how they can apply SPL to their software product. This review tries to study articles in which adoption models and their properties were discussed in order to get a clear perspective about SPLs implementations and adoption strategy. This study is conducted as a Systematic Literature Review which was used to identify important characteristic which should be considered for product line adoption. 22 primary studies from different sources were evaluated to answer 2 research questions. The research identified more than 30 paper on this study but only 21 of them was precisely relevant in the field of SPL adoption. This research provides a general guideline for organization which wants to use SPL in their company. Our preliminary study conclude that in order to choose an appropriate adoption model organizations should identify their needs and choose best way according to that.

Keywords: *Software Product Line, Adoption, Adoption Model, Adoption Strategy*

1. INTRODUCTION

Most of companies provide products for particular customers or markets. Despite of some differences most of them tend to have some common features and form a product family so in order to increase quality, shorten development time and decrease costs, an increasing number of these companies realize that product line development can be useful. One definition for software product line was provided by Software Engineering Institute (SEI): "A software product line (SPL) is a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way"[1]. One other definition is "A software product line (SPL) can be seen as a system family that addresses a specific market segment. In order to improve the productivity and quality of SPL development, a study by Nicolas Anquetil in [2], proposed methods and techniques to motivate the specification,

modeling and implementation of a system family in terms of its common and variable features. Traditional SPL development typically organized in terms of two main processes: in domain engineering, developers try to develop and maintain reusable assets, define the scope and production plan. In this stage developer focuses on scoping, specification and modeling the both common and different features of a SPL, the description of a flexible architecture which include the SPL, common and different features, the production of a set of core assets like components, frameworks, libraries, aspects that addresses the implementation of the SPL architecture. Application engineering, the stage where particular product requirements are gathered, and the product is built by arranging the reusable assets according to the production plan. In application engineering, a feature model configuration is used to create and integrate the core assets produced during the previous stage (domain engineering) so an instance (product) of the SPL architecture can be generated. There are a number of SPL development processes that are



based on this structure [2,3]. Software product line (SPL) is an economic way to develop and maintain the set of variant products in a specific domain [4]. SPL approaches have involved strategic, planned reuse that differs from earlier software reuse paradigms [5]. Changing from a routine system development method, toward product line approach, demand us to adopt new approach. In SPL adoption, an organization tries to change its operational mode to build up product lines consisting of several products instead of developing products separately. During the adoption, organization requires to coordinating and planning technical, organizational, management, and personnel changes [6]. Product line adaption may involve: configuration of system and its component, adding new components to the system, choosing components from an existing components library, modifying components to meet new requirements. Northrop in [1] defined adoption as “moving from some form of developing software-intensive systems with a single-system mentality to developing them as a software product line. The adoption objective is to have a core asset base, supportive processes, and organizational structures, to develop products from that asset base in order to meet business goals, and to institute mechanisms to improve and extend the software product line adoption effort as long as it makes sense [1]. By using software product line, organizations can achieve considerable cost reduction, time to market and simultaneously increase the quality of their software family. However, to date, there are considerable barriers to organizational adoption of product line practices and there is still uncertainty among developers about how it would apply in their own development context. Phased adoption is important because it can be considered as a risk reduction and economically optimizer [1]. There are two types of approach for SPL adoption; one can be made by starting a product line from scratch or by using existing systems [6]. Choosing the appropriate model respect to industry needs, customers and market is probably one of the most challenging steps during the whole process and there is still uncertainty among developers about how SPL would apply. In this research we try to review articles in which adoption models and their properties were discussed in order to get a clear perspective about SPLs implementation. In [1] Adoption Factory pattern is presented by Northrop and provides a roadmap for product line adoption. There is another type of SPL modeling which is being considered. Recently systematic literature reviews focusing on various aspect of software

product line, has been conducted like domain analysis solutions for SPLs, domain analysis tools, requirements engineering, requirements frameworks for SPLs, variability management in SPL. SLR is popular in software engineering because it provides methodologically precise review of conducted studies and is the main method of evidence synthesis. SLR will help researchers to assess, locate and aggregate information from relevant studies [7]. Therefore, in this paper a systematic literature review (SLR) was conducted and studied papers about SPL adoption from 2002 until 2012.

2. METHOD

We undertook this study as systematic literature review and employed methods and steps which are based on Kitchenham’s general guideline for systematic reviews appropriate for software engineering researchers. As part of the process, we formulate a plan for this includes the method which has to be followed, 2 research questions that answering to them is final goal of research and the extracted data. A systematic review consists of several steps, which can be included into three main phases: Planning the review, conducting and reporting. Table 1 shows, three main phases and steps in each one to conduct systematic review [8]. Systematic mapping studies can also be used to categorize the primary studies in research scope. The main difference between these two methods is that systematic literature reviews focus on specific research questions rather than specific topic [9]. Following subsections explain the methods used in order to perform this review.

Table 1: A Systematic Review Phases

- | |
|--|
| <ol style="list-style-type: none">1. Review Planning<ul style="list-style-type: none">• Identify the need for a review• Develop and validate review protocol2. Conducting the review<ul style="list-style-type: none">• Identify primary studies• Primary studies selection• Assess the quality of primary studies• Extract data• Synthesize data3. Reporting the results |
|--|

2.1 Research Questions

The goal of this study is to review different software product line adoption strategy or model for the purpose of achieving higher productivity



and efficiency of the organization. So from this goal we have to answer following questions:

RQ 1: How can an organization adopt SPL for their product family? What are the different ways of SPL adopting?

RQ 2: How can an organization choose the best strategy with respect to their needs? What are the effective factors in selecting one model?

With respect to question 1 we review different ways or models of SPL adopting. To do so we identified the number of related article mostly published during 2002 -2012. Our primarily study scope consists of articles which some conducted SLR methods like [6], some introduces new models or describes existing ones, e.g. In [1] factory pattern was discussed, and rest of them just compare and contrast models.

With respect to RQ2, we considered the scope of the study which looked at different factors of employing of SPL. There is always some barrier to adoption of product line practices and these barriers are somehow determinant the suitable strategy [10]. In order to have successful product line, organizations need adoption strategy with low barriers. They demand for low risk strategy that incurs small upfront effort, swift investment return for incremental transition from current practices [11]. Like economic factors which discuss in [12] by Schmid or other barriers introduced by Krueger[11].

2.2 Search Strategy

In following we explain the strategy that we used in order to provide our primarily study for the systematic literature review.

2.2.1 Data sources

Software product line adoption keywords were searched in journal and conference listed in Tab. 2 We also used the number of search engines for our reviews: ACM Digital Library, IEEE Xplore, SpringerLink, LNCS journal, Google Scholar and also UPM digital library. These journals were chosen because they are well known and have been used as source for other SLR related to software engineering. Chosen journals are listed Table 2:

Table 2: Selected Journals

Source
ACM Communications of the ACM ACM Computing Surveys ACM Transactions on Software Engineering and methodology IEEE Transactions on Software Engineering LNCS journal SEI (Software Engineering Institute) ACM/IEEE Int. Conf. on Softw. Eng. (ICSE) 6th DoD Software Product Line Workshop by SEI IEEE Int. Conf. on Software Product Line, 2011

2.3 Criteria for Data Selection

We define set of inclusion and exclusion criteria which all of our primary studies must evaluated against them then exclude papers which are not relevant to subject.

2.3.1 Inclusion criteria

We study articles about the software product lines adoption and studies with these criteria were included:

- Papers which focus on SPL
- Papers which describe software product line adoption models (RQ1)
- Studies in which different barriers and limitation of adopting was discussed in (RQ2)
- Publications that include effective adoption factors (RQ2)

In the case of reported one study by several publications, we only include the most complete copy.

2.3.2 Exclusion criteria

Studies with following exclusion criteria were leave out from the review:

- Studies which just discussed about SPL not its adoption
- Reviews which discussed SPL usage only
- Studies that its publisher cannot be trusted
- Papers or reports which there is only an abstract or introduction are available for them

2.3.3 Exclusion criteria

Studies with following exclusion criteria were leave out from the review:

- Studies which just discussed about SPL not its adoption
- Reviews which discussed SPL usage only
- Studies that its publisher cannot be trusted



- Papers or reports which there is only an abstract or introduction are available for them

2.3.4 Primary sources selection

The title, keywords and abstract was reviewed in order to select primary sources. At this stage, those primary sources which seemed to be completely irrelevant were only excluded. Then we obtained full text of primary studies and reviewed them against the mentioned above exclusion and inclusion criteria.

2.4 Quality Assessment

For quality assessment included primary studies was only publications from Tab.1. Guidelines in “Systematic literature reviews in software engineering – A systematic literature review” by Kitchenham were used. There are four quality assessment (QA) questions which all the primarily studies should evaluate based on them these questions are:

QA1. Are the review’s inclusion and exclusion criteria described and appropriate?

QA2. Is the literature search likely to have covered all relevant studies?

QA3. Did the reviewers assess the quality/validity of the included studies?

QA4. Were the basic data/studies adequately described? [13]

The questions were scored as Y (yes), N (no) and P (partly)

2.5 Data Extraction

The extracted data from each paper was:

- It’s source (conference or journal) and full reference
- Type of paper (SLR, systematic mapping review, Meta-analysis MA)
- Main topic scope
- Summary of the study including the main research question and the answers [13]
- Do the paper present SLR guideline
- If the paper is SLR, how many primary studies were used

Data was extracted from selected sources to answer mentioned research question..

2.5.1. Primary study data

In order to answer question 1, the following information were extracted from each relevant primary study:

- Various models for adopting SPL
- Different properties of adoption model

We also extracted following information for our second research question from each study:

- Different adopting factors
- Criteria for SPL adopting
- Economic barrier for SPL adoption

3. RESULTS

This section summarizes the results of the study. We identified 21 unique articles by this search process. Other potentially relevant studies that were excluded as a result of applying the detailed inclusion and exclusion criteria are listed above. All studies were published in conference proceedings as well as in journals. Some papers was repeated in two or three library, in these cases the first library was counted as main source e.g. among 4 relevant paper found in ACM library, 3 of them was found in IEEE explore so there is only one paper from ACM.

3.1 Research Question 1

Question 1: How can an organization adopt SPL for their product family? What are the different ways of SPL adopting?

Generally there are two approaches toward SPL adoption. In the first one organization do every step from beginning without using any existing systems, this approach called starting from scratch. Another strategy known as exploiting existing systems, uses existed systems to produce new product line. There are other methods for product line adoption that each has its own advantages in different situations [4].

There are also two different approaches for adopting SPL, evolutionary and revolutionary adoption. In evolutionary approach component and architecture evolve as new product posed, but in revolutionary approach component and architecture match to whole expected product line members.

Evolutionary approach is low risk but has disadvantage that the total amount of investment until the product line components and architecture are totally in place, is larger than using the revolutionary approach.

In proactive approach core asset is developed at first. With these approach organizations can decrease time to market for their products but, it needs upfront investment [10]. Proactive approach is like the waterfall approach to conventional software [11]. In reactive approach we can start from one or more product and then generate core assets and more products. Reactive approach is like

spiral or extreme programming approach to conventional software.

Incremental strategy is a reactive or proactive hybrid which develops parts of core assets then produce one or more products then develop rest of core assets and so on.

In big bang strategy the common and complete platform is built by domain engineering. After platform assets become ready, applications are derived from the platform by application engineering [14].

In [15] Bosch has discussed four organizational models that can be applied during adoption of SPL: Development department, Business units, Domain engineering unit, Hierarchical domain engineering unit.

In development department model development of software is focus on a single development department, not organizational specialization exists with software product line assets or the product family. This model is suitable for smaller organizations. Advantage of this approach is its simplicity.

Business unit model employs a specialization of systems type in the form of business units. Advantage of business unit's model is its ability in sharing of assets effectively. Domain engineering unit is responsible for design, development and evolution of the reusable assets. Developing and evolving the products based on PL assets is, Product engineering unit's responsibility.

Large or very large organizations with wide variety of long-lasting systems usually use Hierarchical domain engineering unit model [15]. Software Product line can also be used in mobile application (hardware and wireless communication technology) and applied to middleware which is designed to provide a set of features in order to meet the needs of multiple problem domains[16]. Software Product Line can be very useful to state the different requirements of devices in terms of commonalities and variability of a middleware platform family. So instead of including common services, only that configuration would be installed which fits the device features.

Based on observation, familiar SPL methods applied in other applications also can be used in middleware development. But most cases arise with the idea of dynamic adaptation, In which production of software is capable of adapting to user needs fluctuations and evolving resource limitations[16].

So far only single adoption models are discussed. Significant improvement has been proved by SPLs when changes in core assets were anticipated with accuracy. In complex or large PL projects, changes in market conditions require combining SPL methods with other alternative methods. Agile software development can be a good alternative because it can cope with changing in customer's needs. Agile software development also needs addition when it comes to efficiently manage reusability and variability in product family. As a result a new method called Agile product line engineering integrating Agile software development with SPLE [20]. According to Carbon [21] combination of Product Line Engineering and agile methods can leads to a higher level of agility compared to engaging only one adoption model in isolation [21].

3.2. Research question 2

Question 2: How should one choose the best strategy with respect to one's needs? What is the effective barrier in selecting one model?

According to our studies choosing suitable model depends on different levels in organizations and organizational structures. Another factor which can influence selection is domain size. There are other determinant factors in SPL adoption like organization goals if time to market is more important than investments then proactive approach can be used [22].

In starting from scratch strategy, organizations can have shorter development time for one product but upfront investments are higher compare to other alternative, known as exploiting existing systems. In this strategy existing systems can be used as much as possible, which cause faster reduction in cumulative costs than in the starting from scratch strategy [23].

There are lots of barriers for SPL adoption and if a company wants success, it must overcome them. Some of them are: SPL adoption is time consuming and will increase time to market, investment in adoption cost, need to have long term plan for product line vision, necessity for change not only in organization development but also in business areas, lack of explicit definition for development process. Software professional only focus on technology and they do not have sufficient knowledge about SPL, lack of enough experts and expensive training, no management rules, no tools for support adoption. There are also other barriers and obstacles which organizations may face during SPL adoption: core team does not have product line



vision, lack of SPL management, asset development practice is not enough [4].

4. DISCUSSION

In this section we discuss about results in the context of mentioned research questions. Number of adoption models was identified in this study which has been applied in SPL context of different organizations. These models can be used independently or in combination with the other models. But, according to our findings there are not much available papers on different combination approach and also few papers discussed about effectiveness of such combination strategies, or what is the most suitable situation for using this strategy. In primary studies there is only one paper which covers this issue [20]. This was mentioned in 3.1.

According to our primarily studies, most of them mentioned barriers in SPL adoption but, in general there are only a few papers which covered this fact like [5],[3],[12],[1] and among them only Krueger in [11] discusses about it specifically. In [23] Bosch discussed about organizational models and advantage and disadvantage of each one. There are not enough empirical studies about role of organizational characteristics on choosing the best strategy. So it is necessary to conduct more empirical studies about these barriers and develop solution for each one. Generally adoption barriers can be a subject for future researches.

The other important limitation in primarily studies was lack of sufficient research on role of management in SPL adoption. As we all know Management is one of the important aspects of successful SPL in organizations, but we cannot find enough papers focused on this subject.

The other limitation in primary studies is that they don't provide enough guidance for organizations which are interested in adopting mentioned methods [7].

4.1 Related works

The literature on SPL Adoption provides some studies which concerning general and specific issues. After reviewing the literature on SPL, we found several sources. Related works which has been done in this area include Böhne in [24] discussed about context of product line adoption like market, organization, business unit and individuals then using these context for choosing the suitable strategy. Böhne also talked about

proactive /reactive approach. In [6] Kuvaja has done a SLR about SPL adoption. Schmid explain about product line economic impact in [12]. Northrop in "Software Product Line Adoption Roadmap" [25] explains factory pattern as an adoption model. Bosch in [23] did research about artifacts of product line and organizational models. There are number of systematic literature reviews in this domain, in [6] SLR was conducted about SPL adoption. In [16], Morais did a systematic review about product lines applied to mobile middleware. Diaz in [20] did a SLR about combining SPLE with agile development model. In [7], Alvez perform SLR about Requirements engineering for software product lines. Chen in [8] discussed evolution of variability management approaches in software product lines.

5. CONCLUSION

This study aims to investigate the up to date strategy for SPL adoption, through systematic literature and specify studied issues in SPL adoption domain. This paper attempts to provide a guideline for researchers who want to plan a future research in this domain. In general experience reports did not provide sufficient information about the processes needed to adopt SPL. In particular they do not present information about potential drawbacks. That is why it would be hard to build an evidence based guide for selecting the most appropriate strategy, generational structure or maturity level for specific situation or context. Hence we need more industrial experiments that report on these factors.

One of our primarily study [1] discussed about SPL adoption roadmap and factory pattern. We try to identify all available approach towards SPL adoption and find the best situation for them to use. We also identify some important barriers in software product line adoption which listed in 3.2 sections (RQ2).

5.1 Implications for Research or Practice

Based on our review, we identified the following implications for research and practice:

More empirical research needed: SPL adoption should be evaluated more thoroughly and repeatedly in empirical studies especially regarding their actual practice.

There are some well known and special types of adoption model which has been used in most companies. There is need for exploring other approach besides existing ones.



REFERENCES:

- [1] Linda M. Northrop, "Introduction to Software Product Lines Adoption Software", *Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213 USA*, IEEE, September 2011, Page(s): 359 – 359
- [2] Nicolas Anquetil, Uirá Kulesza, Ralf Mitschke, Ana Moreira, Jean-Claude., "Model-driven traceability framework for software product lines," *Journal Software and Systems Modeling (SoSyM)* Volume: 9, Issue: 4, 2010, pp: 427-451
- [3] D. Li, "Initiating and Institutionalizing Software Product Line Engineering: From Bottom-Up Approach to Top-Down Practice", *Dept. of Comput. Sci., Iowa State Univ., Ames, IA, USA*, IEEE, Vol 1, September 2009, pp: 53 – 60
- [4] G. Zhang, Sch. of Comput. Sci. & Technol., Fudan Univ., Shanghai, China, "Incremental and Iterative Reengineering towards Software Product Line: An Industrial Case Study", *Sch. of Comput. Sci. & Technol., Fudan Univ., Shanghai, China IEEE*, 2011, pp. 418 - 427
- [5] Eunman Koh., "Issues on adopting software product line," LG CNS Co., Ltd., Seoul, South Korea, 0-7695-2245-9, *IEEE*, 2005, pp. 589
- [6] Kuvaja .P. , Similä J. , Hanna Hanhela." Software Product Line Adoption – Guidelines from a Case Study". *Department of Information Processing Science, University of Oulu, FI-90014 University of Oulu, Finland, LNCS 4980*, 2011, pp. 143–157
- [7] Alves .V. , Niu N. ,Alves C. and Valença G., "Requirements engineering for software product lines: A systematic literature review , *Information and Software Technology 52* , Elsevier B.V, 2010, pp. 806–820
- [8] Chen, Y. and Babar , M.A., "Systematic review of evaluation of variability management approaches in software product lines", Elsevier B.V, 2011
- [9] Catal, C., "On the Application of Genetic Algorithms for Test Case Prioritization: A Systematic Literature Review", *ACM*, 2012, 978-1-4503-1509-8/12/09
- [10] Bastos J.B, Silveira Neto., "Adopting Software Product Lines: A Systematic Mapping Study", *Federal University of Bahia , EASE , IEEE*, 2011, pp:11 – 20
- [11] Krueger, C., "Eliminating the Adoption Barrier", *Software, IEEE*, 2002, vol.19, no.4, pp.29-31, doi: 10.1109/MS.2002.1020284
- [12] Klaus Schmid, Martin Verlage., "The Economic Impact of Product Line Adoption and Evolution", *IEEE SOFTWARE*, 2002 ,page 50 .
- [13] Kitchenham .B , O. Pearl Brereton., 2008. "Systematic literature reviews in software engineering – A systematic literature review" *Information and Software Technology.*, 2008
- [14] Pohl, K. , Böckle , G. and Linden, F. J. v. d. "Software ProductLine Engineering: Foundations, Principles and Techniques." *Sercaucus, NJ, USA: Springer-Verlag*, 2005
- [15] Bosch, J., "Software Product Lines: Organizational Alternatives ", 2002, 0-7695-105
- [16] Morais , Y. , Burity, T. and Elias , G. A., "Systematic Review of Software Product Lines Applied to Mobile Middleware." *Sixth International Conference on Information Technology: New Generations IEEE*, 2009, DOI= 10.1109/ITNG.2009.82.
- [20] Díaz, J., Pérez, J., Alarcón, P.P., Garbajosa, J., "Agile Product Line Engineering , A Systematic Literature Review. Software—Practice And Experience Journal. Special Issue: *Agile Product Line Engineering. Volume 41*, Issue 8, 2011, Wiley InterScience, pp 921–941. DOI: 10.1002/spe.1087
- [21] Carbon, R., Lindvall, M., Muthig, D., Costa, P. "Integrating product line engineering and agile methods: Flexible design up-front vs. incremental design." *APLE '06: 1st International Workshop on Agile Product Line Engineering* , Baltimore, Maryland, USA, 2006
- [22] Yuqing Lin., "A New Parameter for Product Configuration in Software Product Lines." *Second International Symposium*, vol.2, 2009, pp.230-233, doi: 10.1109/KAM.2009.26
- [23] Bosch .J., " Adopting Software Product Lines: Approaches, Artifacts and Organization", *University of Groningen Department of Computing Science*, 2002, SPLC 2 .
- [24] S. Bühne, G. J. Chastek, T. Käkölä, P. Knauber, L.M. Northrop, and S. Thiel. "Exploring the Context of Product Line Adoption". *Proceedings of PFE, ACM*, 2003, pp 19-31
- [25] Linda M. Northrop, "Software Product Line Adoption Roadmap", *TECHNICAL REPORT CMU/SEI*, 2004-TR-022 ESC-TR-2004-022 .