

First Turkish Software Product Line Engineering Workshop Summary

Bedir Tekinerdogan
Bilkent University, Turkey

bedir@cs.bilkent.edu.tr

DOI: 10.1145/2382756.2382778

<http://doi.acm.org/10.1145/2382756.2382778>

Abstract

Software reuse has been a goal of the software community since the early days of software engineering. In this context software product line engineering (SPLE) has gained a broad interest in both academic institutions and industry. This trend can also be observed in Turkey. In the recent years an increasing number of software companies in Turkey have adopted a SPLE approach while others are planning to make the transition. This paper summarizes the results of the First Turkish Software Product Line Engineering Workshop that has been organized in Ankara in June 2012. The primary goal of the workshop was to reflect on the state of practice in SPLE in Turkey. For this five leading SPLE companies in Turkey have shared their experiences in adopting SPLE, and using interactive discussions a research agenda for SPLE in Turkey has been defined. We report both on the experiences from the workshop and the resulting research topics.

Keywords: Software Reuse, Software Product Line Engineering, Workshop Organization, Technology Transfer

1. Introduction

Software reuse has been a goal of the software community since the early days of software engineering [14][11]. Various technologies have been proposed to solve the software reuse problem, including subroutines, object-oriented software development, software design patterns and component-oriented software development. Unfortunately, software reuse has been applied in an opportunistic, ad hoc manner, and as such did not scale up for large-scale software development. Systematic software reuse is a promising approach to reduce cost and development cycle time, improve software quality and productivity. In this context the notion of software product line has gained importance for large scale systematic software reuse. Software product line is a set of software-

intensive systems sharing a common, managed set of features that specify the specific needs of a market segment and that are developed from a common set of core assets [11][20]. Software product line engineering (SPLE) is the process for developing software product lines. Unlike conventional software development paradigms that aim to develop single systems, SPLE considers the development of a family of software systems. As such SPLE adopts a fundamentally different software life cycle approach than single system development.

Currently an increasing number of companies aim to adopt a product line engineering approach with the goal to enhance the quality of products, reduce time-to-market and optimize production costs. The benefits for adopting software product line engineering have been documented by several researchers [6][9][10][11][21] and also indicated from experience reports in practice [7][8][9][16][20][22].

The trend towards systematic software reuse based on SPLE can also be observed in Turkey. Different companies have now decided to apply SPLE approach in order to enhance productivity, increase quality of software, reduce time-to-market and reduce cost. Several national projects between universities and the industry have been carried out in the topic of SPLE.

To reflect on the state-of-the-practice in SPLE in Turkey we have organized the First Turkish Software Product Line Engineering Conference in Ankara in June 2012 [13][23]. During the workshop different SPLE companies in Turkey have presented their experiences regarding the adoption of SPLE and discussed the adopted software reuse practices, the adopted SPLE approach and the obstacles in applying SPLE. In addition to the practical perspective we also aimed to define the important research problems in this context, and as such a list of research questions have been defined from the workshop activities. In this paper, we report on the organization and the outcomes of the workshop.

The remainder of the paper is organized as follows. In Section 2, we provide a short overview on software product line engineering process. Section 3 describes the workshop organization and its goals. Section 4 describes the industrial companies that are developing product lines and that have shared their experiences in the workshop. Section 5 presents the organization of the program. Section 6 describes the topics that were discussed during the workshop. Section 7 provides the list of research questions that we derived from the interactive discussions. Finally section 8 concludes the paper.

2. Software Product Line Engineering

The key motivation for adopting a product line engineering process is to develop products more efficiently, get them to the market faster to stay competitive and produce with higher quality. In alignment with these goals different software product line engineering processes have been proposed such as, the SEI's Framework for Software Product Line Practice [11][19], the Fraunhofer's PULSE-approach [6][22], the Philips' CoPAM method [1], the FAST approach [25], and the Gomaa's PLUS approach [15]. Although different processes have been proposed they share the same concepts of domain engineering, in which a reusable platform and product line architecture is developed, and application engineering, in which the results of the domain engineering process are used to develop the product members.

Very often these approaches are general and need to be customized for the context of the organization. Based on the literature study on the existing SPLE processes we can derive a common SPLE process that seems to recur in different publications. In general there appears to be a consensus that the SPLE process consists of life cycle processes of *domain engineering* and *application engineering*. This common SPLE process is shown in Figure 1.

The domain engineering process is responsible for establishing the reusable platform and thus for defining the commonality and the variability of the product line [20]. The platform consists of all types of software artefacts (requirements, design, realization, tests, etc.). The domain engi-

neering process is composed of five key sub-processes: *product management*, *domain requirements engineering*, *domain design*, *domain realization*, and *domain testing*. The product management process defines the product roadmap which describes the features of all applications of the software product line and categorizes the feature into common features that are part of each application and variable features that are only part of some applications. In addition, the roadmap defines a schedule for market introduction. Based on the output of the product management activities in the *domain requirements engineering* phase the domain requirements are defined that are common to all applications of the software product line as well as variable requirements that enable the derivation of customized requirements for different applications. The domain design will be responsible for designing the product line architecture which represents the common architecture for the products in the selected product line. Based on the product line architecture the required product line artefacts are developed and stored in a reusable asset base. The domain engineering process also includes the phase domain testing which results in the domain test plan, the domain test cases, and the domain test case scenarios.

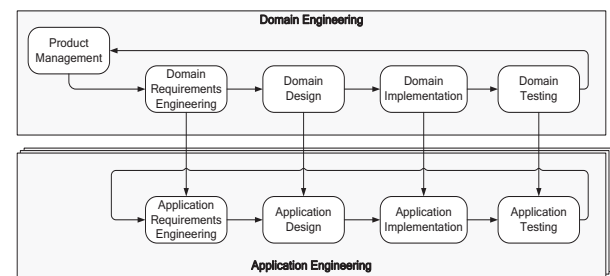


Figure 1. General SPLE Process

In the application engineering process the applications of the product line are built by reusing the artefacts and exploiting the product line variability as defined in the domain engineering process. The application engineering process is composed of the sub-processes *application requirements engineering*, *application design*, *application realization*, and *application testing*. The application requirements engineering reuses the domain requirements and defines the requirements for the particular application. The application design process takes as input the application requirements and by reusing the product line architecture the application architecture is developed. Application design selects and configures the required parts of the reference architecture and incorporates application specific adaptations. The application realization sub-process creates the considered application. The main concerns are the selection and configuration of reusable software components as well as the realization of application-specific assets. Reusable and application-specific assets are assembled to form the application. The application testing sub-process comprises the activities necessary to validate and verify an application against its specification. The application testing process takes as input all kinds of application artefacts to be used as a test reference, the implemented application, and the reusable test artefacts provided by domain testing. The output comprises a test report with the results of all tests that have been performed.

3. Workshop Organization and Goals

The workshop [13] has been organized by the Bilkent Software Engineering Group, Bilkent University [4]. The workshop was affiliated with the 6. Turkish Software Engineering Conference that was held in June 2012 at Hacettepe University, Ankara, Turkey [23]. The workshop organization was initiated as part of the ongoing research and consultancy activities between the Turkish software industry and the universities. The participants of the workshop included software engineering researchers and participants from leading companies in Turkey which apply software product line engineering.

The purpose of the workshop was to bring together software engineering practitioners and researchers from industry and academia in Turkey to

exchange experiences, results and ideas related to software product line engineering concepts. As a first workshop in this field in Turkey, our particular goal of this workshop was the following:

- *Reflect and foster state-of-the practice of SPLE*

With this workshop we hoped to reflect the state-of-the-practice with respect to SPLE in Turkey. So show the benefits and risks of SPLE in particular experiences from real industrial projects is important. As such, the workshop aimed to provide an opportunity to represent the latest developments in industrial software projects and highlight the identified problems and the solutions. Experiences that were shared from the leading software companies would help to provide a better insight in the topic of SPLE for both academics and practitioners.

In addition to the above main goal we also hoped to achieve the following goals:

- *Stimulate research and education on SPLE*

We hoped to stimulate the research and education in Turkey with respect to SPLE. For researchers we aimed to find a forum and a channel to present and share their ideas. Educators would find the important topics in SPLE and include these in their courses. The first software product line engineering course was started in 2008 at Bilkent University in Ankara [24]. With the workshop we aimed to further trigger the interest of educators for the SPLE topic.

- *Support MSc and PhD students in providing directions guidelines for their research*

The workshop would provide an opportunity for PhD students who are doing research on SPLE to identify the key obstacles in SPLE and form their own research agenda.

Given the large number of participants and the active involvement of the participants during the workshop we believe that this first national SPLE workshop has been very useful to support these goals. A survey among the participants showed that there was also a clear interest in the second SPLE workshop.

4. Participants

We have invited 7 leading software companies in Turkey to the workshop to hold a presentation. These companies were selected because of their active involvement in SPLE projects. Two of them indicated that they just had started adopting SPLE and that it was too early to share their experience. The other 5 companies that we invited all happily accepted the invitation and prepared a one page abstract and the corresponding presentation. The one page abstracts (in Turkish) have been published in the conference proceedings of the 6. Turkish Software Engineering Conference.

We shortly describe the companies that presented during the workshop in the following:

- **ARÇELİK**

Arçelik A.Ş. [2] is a household appliances manufacturer in Turkey which products include white goods, electronic products, small home appliances and kitchen accessories, such as refrigerators, freezers, washing machines, dishwashers, aspirators, vacuum cleaners, coffee makers and blenders. Arçelik A.Ş. is active in more than 100 countries through its 13 international subsidiaries and over 4500 branches in Turkey. The Company operates 10 production plants in Turkey, Romania and Russia, including refrigerator, washing machine, dishwasher, cooking appliances and components plants. Further, it offers products under its own ten brand names, including Arçelik, Beko, Grundig, Altus, Blomberg, Arctic, Defy, Leisure, Arstil, Elektra Bregenz and Flavel. The company aims to apply SPLE for the refrigerator software product family. In particular, it aims to combine this with model-driven development approach for supporting the code generation in the application engineering process.

- **ASELSAN**

ASELSAN is a Turkish corporation that was founded by the Turkish Army Foundation in 1975. The company develops tactical military radios and defense electronic systems for the Turkish Army [3]. Since 1976 SELSAN has expanded its product and customer portfolio, and has now become a leading electronics and electronic systems company in Turkey that designs, develops and manufactures modern electronic systems for military and industrial customers, in Turkey and abroad. The company headquarters is situated at Macunköy facilities in Ankara, Turkey. Currently, ASELSAN has been organized in four main divisions:

1. Communications Devices Division (HBT),
2. Defense Systems Division (SST),
3. Radar, Electronic Warfare and Intelligence Systems Division (REHIS),
4. Microelectronics, Guidance and Electro-Optics Division (MGEO).

In all divisions, methodologies complying with military standards and ISO-9001 are successfully applied using computer aided design (CAD), computer aided engineering (CAE) and computer aided manufacturing (CAM) technologies. For the workshop we have invited the SST and the REHIS groups which are independently managing SPLE projects.

- **CYBERSOFT**

CYBERSOFT has been established in 1995 and focused on development for IT projects employing advanced information technologies and application development based on the object oriented approach [5]. In particular, CYBERSOFT aims to create a broad vision concerning the development of governmental information technologies. The headquarters of Cybersoft is located in Ankara with a large division in Istanbul. CYBERSOFT has completed several large-scale public sector projects successfully. Recently the company has focused on widening its development and for adopting global software development. In addition the company aims to benefit from large scale systematic software reuse as defined by SPLE.

- **MILSOFT SOFTWARE TECHNOLOGIES**

MilSOFT is a system integration and software development company, having business presence and interest in defense industry [18]. The main interest areas are C4I, Data Links and Messaging, Image Exploitation Systems, Electronic Warfare, Embedded Systems and HW Manufacturing Through Subcontract Management. The software engineering process used in MilSOFT has recently been adapted to CMMI (Capability Maturity Model Integrated) Level 5 requirements. Milsoft has adopted large scale software reuse using application frameworks for the corresponding domains. It aimed to enhance software reuse and increase ROI through transitioning to SPLE.

5. Program

The program of the workshop was organized as follows:

9:00-9:45 Introduction to Workshop and Software Product Line Engineering

Here we shortly presented the SPLE concepts and the corresponding process. In addition the program of the workshop was announced.

9:45-10:30 Cybersoft – Software Product Line engineering within the context of Global Software Development Projects

The presentation discussed the experiences of reuse within global software development projects related to banking and insurance applications. Reuse on single site has its own challenges but if the aim is to provide systematic software reuse based on SPLE in such global settings then the challenges multiply a lot. The presentation presented the strategy for coping with these challenges and discussed the different architecture design alternatives. Further, it was shown that the need for declarative languages was necessary to achieve the SPLE reuse goals within global software development.

10:45-11:30 Milsoft ICT, From framework to SPLE approach for Command and Control Systems

The presentation discussed the general domain for command and control systems and the need for systematic software reuse. The company presented the framework which seemed to be quite successful. Due to several obstacles and the vision for optimizing software reuse it was decided to enter SPLE process. The presentation presented the first results and the adopted transition process together with the experienced problems.

11:30-12:15 Aselsan-SST, Experiences from 2 SPLE projects within the context of defense systems

The presentation discussed the experiences in setting up two different SPLE projects within the same company. One project was about SPLE for “Weapon Control System Product Family”ç Here the Feature-Oriented Reuse Method (FORM) was adopted as a product line engineering approach. The other was “Self Protection and Fire Support Product Family” in which the SEI’s approach for SPLE was used. The presentation discussed organizational issues, the adopted process, the important benefits of SPLE and the obvious ROI of SPLE for the company.

12:15-13:00 Arçelik, SPLE approach for Refrigerator Software Product Family

The presentation discussed the transition process for adopting SPLE for refrigerator product family. The main domain of refrigerator product family, the low time-to-market in this domain and the stringent competition was discussed as the setting of the project. To meet the time-to-market constraints both an SPLE approach and model-driven development approach was envisioned. The main focus in the presentation was the domain modeling for the dynamic behavior of refrigerator product family. Based on this the application engineering, i.e. code generation, was automated. The presentation discussed the problems and the future development plans.

14:00-14:45 Aselsan-REHIS, An incremental SPLE process for Multiple Product Line Engineering

In general there seems to be a common agreement that product line engineering is not a pure sequential design process in which each activity is completed and then followed by subsequent activities. Rather, software product line engineering usually requires repeating the activities (iterative) and developing the artefacts in smaller portions at a time (incremental). Unfortunately, the proposed product line engineering processes seem to largely remain silent about the need for incremental and iterative nature of the product line engineering processes, or do not discuss this in detail. In this presentation experiences in realizing such an incremental and iterative product line engineering process within the industrial context of Aselsan was presented. The presentation discussed the needs for the iterative and incremental PLE process from the business and organizational perspective, and described the design of the process as well as its application within the context of Aselsan.

14:45-15:30 Preparing Discussions

The audience was split up in 5 small groups for discussing the research topics. This is further explained in section 6.

15:45-17:30 Interactive Session-Discussion

6. Workshop Topics

Since the workshop was focused on sharing experiences of leading software companies in Turkey we did not constrain the topics for the workshop presentations. However, to guide the companies in preparing their presentations we asked them the following questions:

- What is the level of reuse within the company?
- What have been the reuse techniques that were adopted?
- Why did you decide for SPLE approach?

- What is the domain and the product line that was considered? What defined the product line scope?
- What was the adopted transition process for SPLE?
- What were the main lessons learned during the transition process?
- What were the main lessons learned during the actual execution of the SPLE process?
- What was the return-on-investment for the company?
- Any other experiences or lessons learned?

By asking these questions beforehand we hoped to somehow direct the presentations towards the SPLE context, without unnecessarily constraining any other related topics. In the end, the topics that were addressed during the workshop presentations and the discussions were the following:

- Software production line transition strategies
- Software Product Management
- Aligning organization for SPLE
- Software Product Line Requirements Engineering
- Software Product Line Scoping
- System Product Line Engineering
- Software Product Line Architecture
- Model-Driven Product Line Engineering
- Automating SPLE
- SPLE within Agile Context
- SPLE within Global Software Development
- Software Product Line Tools
- Metrics for calculating ROI

The main issue that was in particular discussed was the ROI for the organization and the organizational requirements for adopting an SPLE process. One of the overall conclusions was also that SPLE should not be just considered as a technical process but requires insight in business, organizational and managerial processes.

7. Result of Discussions

Organization of Discussions

The second part of the workshop included an interactive session in which we focused on the potential research questions that were derived from the experiences of the software product line companies. To provide a systematic guidance to the workshop activities we adopted four steps. In the first step, we split up the audience into five different groups. The groups were formed so that it included participants from different companies and universities. In the second step, we started the elicitation of research problems within each group. To involve each participant in the discussion we made use of index cards that were distributed to all participants. The task of each participant was to write down at least five research questions and/or obstacles that they thought need to be solved for adopting a successful SPLE approach. In the third step of the discussion session, the group had to collect all the index cards, categorize the resulting research questions and select the final set of questions. The fourth step of the discussion session included presentations per group. For this each group representative discussed the categories of research problems and the final outcome of the set of research problems. Two or three groups also gave the prioritization of the research problems.

Identified Research Problems

The identified research problems were as follows:

- How to characterize an organization and devise the proper transition process?
- How to identify and cope with organizational obstacles including the alignment of the organizational structures.
- How to apply SPLE in customer-centric organizations?
- How to migrate legacy system to SPLE system
- What are the patterns that should not be used when adopting SPLE, i.e. Software Product Line (Anti)-Patterns?

- How to manage the evolution of SPLE projects?
- How to deal with Configuration Management in SPLE and multiple product line engineering?
- What are the best practices for model-driven SPLE? What are the required MDD patterns?
- What parts of SPLE can be automated? Which parts should preferably not be automated? How to combine with legacy code?
- What are the challenges in developing DSLs for SPLE? How to decide on the domains and the corresponding DSLs?
- How to design/optimize SPLE Architecture for quality (e.g. reuse, reliability, sustainability..) besides of functional features?
- How to keep the SPLE project sustainable? What are the potential risks, how to mitigate these risks?
- What is the right scope of the SPLE architecture?
- How to do testing in SPLE? How to cope with the challenges?
- How to cope with scalability of variability modeling (e.g. feature modeling)
- How to adopt different reuse approaches/mechanisms in SPLE?
- How to traceability between feature models and SPLE assets
- How to do integrate SPLE in agile approaches? What are the challenges?
- How to cope with hardware/software co-design in SPLE?
- How to cope with impact of software evolution in SPLE?
- How to design SPLE assets for quality (e.g. testability)
- How to define multiple Views for domain modeling in general and variability modeling in particular?
- What are the important metrics for SPLE and how to measure progress of SPLE?
- How to organize product lines within multiple product line engineering.

8. Conclusion

In this paper, we have reported on the First Turkish Software Product Line Engineering Workshop (TSPLE). The workshop was initiated as a result of the active projects on SPLE in the Turkish software industry. The primary goal of the workshop series was to reflect on this state of the practice and likewise foster the research and education in the field of SPLE in Turkey. The workshop turned out to be a big success regarding the interest that it received from both the academic institutions and the Turkish software industry. There were around 60 participants, 65% of these were from industry. For a first workshop we believe that this can be considered indeed a success. The five companies that have presented during the workshop have all provided a unique insight in the topic of SPLE in general. We have seen the experiences of applying SPLE within an agile environment (ASELSAN-REHIS), SPLE within global software development (CYBERSOFT), experiences of managing different SPLE projects together (ASELSAN-SST), multiple product line engineering (ASELSAN-REHIS), evolution of framework to SPLE (MILSOFT), and model-driven development approaches within SPLE (ARÇELİK). In addition to the lessons learned a list of important research questions has been identified. We believe that SPLE will further evolve in Turkey in the recent years. In our future work we will continue our collaborations and organizations of events on this topic.

Acknowledgments

We would like to thank the industrial participants and presenters of the workshop: İbrahim Niyazi Ülgür and Özkan Akgül (Arçelik), Özgü Özköse Erdoğan and Onur Aktuğ (Aselsan-Rehis), Tolga İpek and Ertağrül Barak (Aselsan-SST), Semih Çetin (Cybersoft), Turgay Çelik and Ekrem Serin (Milsoft). Further, we would also like to thank all the other participants who have attended the workshop.

References

- [1] P. America, H. Obbink, R. van Ommering, F. van der Linden., Co-PAM: A Component-Oriented Platform Architecting Method Family for Product family Engineering, 167-180. Proceedings of Software Product Lines First International Conference. Denver,

- Colorado, August 28-31, 2000. Boston, MA: Kluwer Academic Press, 2000.
- [2] Arçelik, <http://en.wikipedia.org/wiki/Arçelik>, 2012
- [3] Aselsan, <http://www.aselsan.com/>, 2012.
- [4] Bilkent Software Engineering Group, <http://www.cs.bilkent.edu.tr/~bedir/Bilsen/>
- [5] Cybersoft, <http://www.cs.com.tr/en/cybersoft.html>
- [6] J. Bayer et al., PuLSE: A Methodology to Develop Software Product Lines. Proc. Symposium Software Reusability (SSR 99), pp. 122–131, ACM Press, 1999.
- [7] A. Birk, G. Heller, I. John, K. Schmid, T. von der Massen, K. Müller, Product Line Engineering: The State of the Practice. IEEE Software. Vol. 20, No. 6, pp. 52-60, November, 2003.
- [8] G. Boeckle, J. Bermejo, P. Knauber, C. Krueger, J. Leite, F. van der Linden, L. Northrop, M. Stark, and D. Weiss; Adopting and Institutionalizing a Product Line Culture, In: Proceedings of the 2nd International Conference on Software Product Lines (SPLC-2), San Diego, USA, Springer, Berlin Heidelberg New York, LNCS 2379, 2002, pp. 48–59.
- [9] G. Böckle, P. Clements, J.D. McGregor, D. Muthig, & K. Schmid, K. Calculating ROI for Software Product Lines. IEEE Software 21, 3, pp. 23-31, June 2004.
- [10] J. Bosch. Maturity and Evolution in Software Product Lines: Approaches, Artefacts and Organization, 257-271. Proceedings of Software Product Lines Second International Conference. San Diego, California, August 19-22, 2002. New York, NY: Springer-Verlag, 2002.
- [11] P.C. Clements, L. Northrop. Software Product Lines: Practices and Patterns. Boston, MA: Addison-Wesley, 2002.
- [12] Cybersoft, <http://www.cs.com.tr/en/cybersoft.html>. 2012
- [13] First Turkish Software Product Line Engineering Conference, Ankara, Turkey, June, 2012.
- [14] W.B. Frakes and K. Kang. Software Reuse Research – Status and Future. IEEE Trans. On Software Engineering, Vol, 31, No. 7, pp. 529-536, July 2005.
- [15] H. Gomaa. Designing Software Product Lines with UML- From Use Cases to Pattern-Based Architectures, Addison-Wesley, 2005.
- [16] F. van der Linden, K. Schmid, E. Rommes. Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering, Springer, 2007.
- [17] J.D. McGregor, S. Jarrad, L.M. Northrop, and K. Pohl. Initiating Software Product Lines, IEEE Software, vol. 19, no. 4, July 2002, pp.24–27.
- [18] Milsoft, <http://milsoft.com.tr/>, 2012
- [19] L. Northrop, et al. A Framework for Software Product Line Practice. Version 4.1, <http://www.sei.cmu.edu/plp/framework.html>
- [20] K. Pohl, G. Böckle, F. van der Linden. Software Product Line Engineering – Foundations, Principles, and Techniques, Springer, 2005.
- [21] K. Schmid, M. Verlage. The Economic Impact of Product Line Adoption and Evolution. IEEE Software, Vol. 19, No. 4, July/August 2002, 50-57.
- [22] K. Schmid, T. Widen. Customizing the PuLSE™ Product Line Approach to the Demands of an Organization, 221-238. Proceedings of the 7th European Workshop on Software Process Technology, (EWSPT'2000) (LNCS 1780). Kaprun, Austria, February 21-25, 2000. New York, NY: Springer-Verlag, 2000
- [23] Sixth Turkish Software Engineering Conference, <http://www.uysm.org.tr/>, Ankara, Turkey, June, 2012.
- [24] B. Tekinerdogan. Software Product Line Engineering Course at Bilkent University, <http://www.cs.bilkent.edu.tr/~bedir/CS415-SPLE/>, 2012.
- [25] D.M. Weiss, C.T.R. Lai. Software Product-Line Engineering: A Family-Based Software Development Process, Addison-Wesley, 1999.