What can Software Engineering Do for Sustainability: Case of Software Product Lines

Ruzanna Chitchyan
Department of Computer Science
University of Leicester
Leicester, UK
Email: rc256@le.ac.uk

Joost Noppen
School of Computing Sciences
University of East Anglia, UK
Email: j.noppen@uea.ac.uk

Iris Groher
Johannes Kepler University Linz
Linz, Austria
Email: iris.groher@jku.at

***Abstract*—Sustainable living, i.e., living within the bounds of the available environmental, social, and economic resources, is the focus of many present-day social and scientific discussions. But what does sustainability mean within the context of Software Product Line Engineering (SPLE)? And what does SPLE do for sustainable living? In this paper we take the first step towards identification of the sustainability-related characteristics relevant to SPLE. The paper also discusses how the key areas of interest to the current SPL community (as reflected by what is measured and optimised in SPLs today) relate to these sustainability characteristics.**

I. Introduction

Sustainability in general is defined as the *capacity to keep up*. While there is no universally agreed upon interpretation of sustainability, within the context of human society, this refers to ”keeping up” the preferred lifestyle, but accommodating it within the environmental, societal, and economic boundaries of available resources. Since software has become one of the cornerstones and main drivers for change in our society, it should both adhere to this notion of sustainability, and promote it. Yet, at present there is no clear understanding as to how such adherence should be accomplished or promoted.

To address this shortcoming, the topic of sustainability in software engineering has recently been considered within the context of Requirements Engineering (e.g., RE4SuSy workshops), Human Computer Interaction (e.g., HCI and Sustainability workshops at CHI), and general Software Engineering community (e.g., GREENs workshops at ICSE and Sustainability workshops at Modularity). There is also a Journal on Sustainable Computing (by Elsevier) and a few conferences, e.g., IEEE International Green Computing Conference and ICT4S.

But what does sustainability (both in technical and broader sense) mean within the context of SPLE? How does it relate to the current key points of interest to the SPL community? Can sustainability be promoted through SPLE? These are the main questions addressed in this paper.

The remainder of the paper is structured as follows. In Section II we introduce the DiVA project used as a basis for our discussion on sustainability in SPLE. Section III discusses what notions of sustainability arise for the DiVA case study in the context of SPLE. Section IV discusses how the current work on SPL relates to sustainability. To do this we look at what concerns the SPL community is interested in measuring and optimising at present and how these concerns relate to sustainability topics we found in the case study, as discussed in Section III. Some related work is presented in Section V. We conclude our paper in Section VI.

II. Study Outline

In order to elicit an understanding of how the topics of sustainability relate to SPLE we study a specific example.

A. Case Study: DiVA Project

The example selected was that of the DiVA project [1], which focused on the application of dynamic variability and aspect-oriented development principles in product lines. The choice of this example was motivated by two factors:

1) In previous discussions on sustainability many of our colleagues from the SPLE community had expressed the view that sustainability is only relevant to customers, stating that ”If the customers do not ask for sustainability, we cannot do anything about it.”. The case study we selected does not have explicit requirements for sustainability. In fact, it is a purely technology-focused project. Thus, if sustainability is to be found relevant here, we can dispel the (rather prevalent) ”only the customer drives sustainability” argument.
2) The documents used in this example are publicly available. Moreover, other than being technology-focused, this case study has no unique characteristics with regards to sustainability, so any other case study would have been equally suitable.

B. Study Method: Grounded Theory Analysis

We did not want to shoehorn the case study into an existing framework of sustainability dimensions, and so began with no pre-conceived expectations on sustainability notions relevant to this case study. Thus, the Grounded Theory (GT) analysis technique [2] was well suited for the task of concept identification. (Details of GT are out of scope in this paper and can be seen in [2].) This technique is widely used in social sciences for text analysis. It requires for a researcher to read and identify (task referred to as coding in GT) as many concepts in the text as he/she finds relevant. Simultaneously, notes on the relevance and interrelatedness of the coded concepts are collected. These concepts are then reviewed, with groups of concepts merged and/or refined, relationships between groups defined. This review aims to produce a conceptual model for the given case study (akin to an ontology for it).

Since the text coding process can often be rather subjective, more than one researcher is engaged with this task. Each carrying out their own coding and validating their colleague's work. Where disagreements arise, the researchers must resolve these and come to a unified code set.

Thus, three researchers (the co-authors of this paper) carried out coding and validation of the relevant concepts in the present case study. We initially defined four coding categories (economic, technical, social, and environmental sustainability). To mitigate threats to validity, the definition of the initial categories was performed using the so-called 4-eyes principle (i.e., defined and checked by at least 2 researchers independently). The category system has been iteratively extended and refined by subcategories during coding (deductive categorization). Each important text fragment was assigned to one or more (sub) categories. If an important text fragment could not be assigned, a new category was created. Newly emerging categories were always discussed by the three researchers. This means that the final category system has actually been developed incrementally through multiple feedback loops during coding. The final category system is reflected in the feature model presented in Fig. 1. We now look at some of the concepts identified in this study in more detail.

III. What Are Sustainability Concepts in SPL: Cases of DiVA Project

Below we briefly discuss the concepts related to sustainability that emerged from our Grounded Theory analysis of the DiVA case study (shown in Fig. 1). We also provide a few extracts form the DiVA text to illustrate the relevant concepts.

A. Economic Sustainability

Economic sustainability is the ability of a business to "keep up" its operation successfully. An SPL is a business model. As all business goes, the model is adopted if a company considers it relevant and maintained for as long as it contributes towards the business profitability. Thus, it is not surprising that such notions as "efficiency" (e.g.,"Efficiency: Does work package 1 (WP1) simplify the analysis of large requirements...") and "productivity" (e.g.,"...in which measure does it reduce the time needed..." or "... the productivity has been evaluated...") come up in the case study analysis.

Yet, we note that a number of issues related to the technology, such as "tool support", "usability", "scalability" have also been identified as relevant to the notion of economic sustainability. This, indeed, is not surprising either as all these tools and their characteristics are adopted by a business to boost its economic interest. Thus, there is a clear link between the economic and technical characteristics of the sustainability.

B. Technical Sustainability

Here technical sustainability relates to the ability of the business to "keep up" its technical assets and their characteristics which ensure useful service of these assets in present and future. Thus, such notions as support for adaptation (e.g., "Specification of adaptation policies: Identifying variability and modeling it efficiently is a key concern when engineering large and complex systems"), heterogeneity and distribution (e.g., "Does the reconfiguration technology support technology heterogeneity and distribution?"), integration (e.g., " Can the technology be integrated with other Thales tools using standard models"), under the umbrella of evolution emerge. Moreover, since software is a core part of SPL, the notion of using (software) tools to produce new (software) tools, i.e., tools support, is also essential (e.g., "... the tools and methodology provided by DiVA enable and ease the design of complex adaptive systems...").

Furthermore, since the chosen case study is very much technically-focused, we observe that the concepts of technical sustainability constitute the largest group as shown in Fig.1.

C. Social Sustainability

As shown in Fig. 1., in the present study we have viewed the concerns related to the Organizational and Personal well being as separate groups, though many consider these two as a parts of the larger Social Sustainability group.

In this case study we observe a relative sparsity of concepts related to social topics. Yet, even here we find several references, such as, notions related to organizational sustainability coded as "ease of tool adoption" (e.g.,"Verifying the ability for engineers to adopt the methodology and the tools"); or to employee sustainability "usability" and "tool support", where tools "...ease the design of complex adaptive systems...". As noted above, the choice of this particular document for concept identification has resulted in sparsity of relevant concepts on this topic. Yet, we must note that there is a substantial body of work within SPL that focuses on process improvement and organizational management under SPL business model. We are confident that, had we chosen a process focused piece for analysis, the conceptual results would be much richer in this category.

D. Environmental Sustainability

In the present case study we have not directly picked out any concepts addressing environmental sustainability. This is because this case study did not directly discuss the topics of resource use reduction or environmental improvements. However, a number of above identified factors could have potentially substantial environmental impact. For instance, increase in productivity and efficiency will often imply reduced resource use; improved technical assets will require less energy and human resources used for their maintenance, better tool support will also quicken development of new systems - again reducing energy and other resource use for a given job. Nevertheless, since we are not looking into the indirect effects of identified concerns, we have not discussed these any further.

IV. Relating Sustainability to Topics of interest in SPL through Metrics Review

Having identified a set of topics related to sustainability in the DiVA case study, we questioned the relevance of these topics to the SPL community. We observed that if an issue is considered to be of relevance to the SPL community, there will be metrics developed for it - we only measure what we are interested in. Thus, we (very briefly) review the metrics currently in use in SPL and note how they relate to the Sustainability concerns. Through this exercise we also identify areas of sustainability that have not been considered as key topics in SPL so far, at least these have not been interesting enough to have them measured and monitored.

Metrics Related to Technical Sustainability

Within SPL development traditional software metrics are used to assess the quality of the software [3], [4], such as (i) number of lines of code, (ii) cyclomatic complexity, (iii) depth of inheritance. These are complemented with metrics specifically tailored for measuring the technical ability of the SPL to support the production process, such as: (i) core asset utility, (ii) percent reuse, (iii) specialised SPL maintainability metrics [5].

All these metrics clearly relate to the ability of the technical assets to "keep up" and provide useful service, i.e., their technical sustainability. With respect to the DiVA case study (as per Fig. 1), the metric of cyclomatic complexity directly measures the *Complexity* concern; lines of code measures *Implementation*, and depth of inheritance relates to *Understandability*; moreover, all these closely relate to *Scalability* and *Evolution*.

Metrics Related to Economical Sustainability

The main goal of SPL business model lies in reducing the business cost and increasing it's productivity. This goal does, indeed, coincide with that of the Economic Sustainability.

The currently used metrics that focus on SPL performance include [6], for instance (i) total product development cost, (ii) time to market, (iii) market feature coverage.

Other metrics measure a more indirect contribution focusing on how the development process itself has been streamlined. The underlying assumption for these metrics is that an improved production process will drive production costs for individual software products down, which in turns repays the investment for creating the streamlined production process. These include, e.g., (i) effort to produce core assets, (ii) core asset utility, (iii) percent reuse.

All these metrics directly relate to the topics of *Efficiency*, *Productivity*, and *Useability*, as depicted under the Economic Sustainability concern in Fig. 1.

Metrics Related to Social Sustainability

We review the metrics related to organizational and personal aspects of SPLs separately.

The SPL metrics related to the organisational aspect seem to focus on such issues as process compliance, or revert back to issues closely lined with the financial gain [5].

On the one hand, this correlation is logical as a process support is essential for a smooth operation of an organisation, and reduced costs in software development translate in continuity of the organisational existence. The topics that have emerged from the DiVA study as relevant to the Organizations sustainability are those of Employee Support (e.g., help the employees to learn to use new tools) and Tool Support (e.g., provide tools that support the work process). These topics relate to the process compliance metrics of the SPL.

On the other hand, such topics as motivation, innovation, cooperation, and trust between colleagues are missing from the set of SPL metrics. These are the topics related to the human side of the organization, which, incidentally, are also amiss in the DiVA case study results (see Fig. 1). These then, are open questions to be addressed by SPL community.

Turning to the metrics related to the Personal sustainability, we observe that these are often centered around the challenges faced by the developers in utilising the SPL infrastructure. This same observation emerged from the DiVA case study, where the individual were considered as Employees and related to such topics as Efficiency, Performance or Tool Support.

In truth, such metrics and topics fall in either the economic or technical areas and are focused on making human capital or an enterprise more productive. Here too, the human side of the employees is missed, with such issues (to name a few) as personal job satisfaction, self worth, or employee equality are missed out entirely.

Metrics Related to Environmental Sustainability Most notably the area of environmental sustainability has received very little attention within the SPL community. Neither did we identify any concepts directly related to the environmental concerns in the DiVA case study.

Nevertheless, as noted above, SPL does have direct impact on the environment, e.g., through resource consumption, or through work process change.

When examining metrics that specifically address environmental sustainability of software development, such as issues as energy efficiency of software [7] or impact of software architectural choice on resource consumption and emissions [8] have been considered. It is clear that similar issues would directly relate to the SPL community, and should be further researched on within the SPL context.

V. Related Work

There is only little work on how sustainability and software product line engineering fit together [9] [10].

Lutz et al. [9] adopt techniques from SPLE for the design and operation of long-lived, sustainable systems (LSS). LSS have an extended lifetime, make efficient use of resources, and are highly adaptable. This work deals with knowledge preservation during system changes with the help of SPLE. Voyager Spacecraft is used as an example of an LSS. Lutz et al. focus mostly on technical sustainability.

Savolainen et al. [10] discusses how SPLs can be built in a sustainable way. They propose a model of planned staged investments with two phases (investment and harvesting) that ensures long-living product lines. Savolainen et al. mainly focus on evolution and technical sustainability of product lines but also discuss how the organization can deal with switching between the two phases.

Other work mainly focuses on how sustainability affects software systems in general [11] [12] [13]. Durdik et al. [11] present a catalog of software sustainability guidelines for reaching the goal of economic sustainability during system evolution. Seacord et al. [12] present measures to evaluate sustainability of software systems. Sustainability in this case focuses on software maintenance and evolvability. Koziolek [13] assesses the capabilities of existing architecture evaluation methods with respect to their support for measuring the sustainability of a software architecture. Again, sustainability is limited to evolvability and maintenance.

VI. Conclusion

In this paper we have undertaken a text analysis of an SPL-based case study and identified a number of sustainability-related concepts relevant to SPLE. We have shown (both through the DiVA case study and wider SPL metrics analysis) that the SPL community has long engaged with topics directly focused on economic, technical, as well as organizational sustainability (irrespective of the specific project domain). We observe existence of a number of metrics relevant to economic and technical sustainability, yet, these are used in silos, not considered as measuring sustainability as a whole. Moreover, topics related to environmental and personal sustainability as well as the human aspects of organizational sustainability have, so far, been largely ignored. These are the topics that we hope to research on further.

References

[1] Diva project. [Online]. Available: https://sites.google.com/site/divawebsite/home

[2] B. Glaser and A. Strauss, *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Aldine Transaction, 1967.

[3] N. E. Fenton, *Software Metrics: A Rigorous Approach*. Chapman & Hall, Ltd., 1991.

[4] Z. Ahmed, "Towards performance measurement and metrics based analysis of PLA applications," *CoRR*, vol. abs/1007.5127, 2010.

[5] E. Bagheri and D. Gasevic, "Assessing the maintainability of software product line feature models using structural metrics," *Software Quality Journal*, vol. 19, no. 3, pp. 579–612, 2011.

[6] D. Zubrow and G. Chaslek, "Measures for software product lines," Software Engineering Institute, Tech. Rep. CMU/SEI-2003-TN-031, October 2003.

[7] F. H., "Near-Optimal Energy-Efficient Joint resource Allocation for Multi-Hop MIMO-AF Systems," in *PIMRC'13*. IEEE, Sep. 2013, pp. 943–948.

[8] R. Chitchyan and et. al., "Study of Architectural Impact On Software Sustainability," in *RE4SuSy'14*, Aug. 2014, pp. 13–16.

[9] R. Lutz and et. al., "Software product line engineering for long-lived, sustainable systems," in *Software Product Lines: Going Beyond*, ser. LNCS. Springer, 2010, vol. 6287, pp. 430–434.

[10] J. Savolainen and et. al, "Long-term product line sustainability with planned staged investments," *Software, IEEE*, vol. 30, no. 6, pp. 63–69, Nov 2013.

[11] Z. Durdik and et. al, "Sustainability guidelines for long-living software systems," in *Software Maintenance (ICSM), 2012 28th IEEE International Conference on*, Sept 2012, pp. 517–526.

[12] R. Seacord and et. al, "Measuring software sustainability," in *ICSM'2003*, Sept 2003, pp. 450–459.

[13] H. Koziolek, "Sustainability evaluation of software architectures: A systematic review," in *QoSA-ISARCS'11*. ACM, 2011, pp. 3–12.

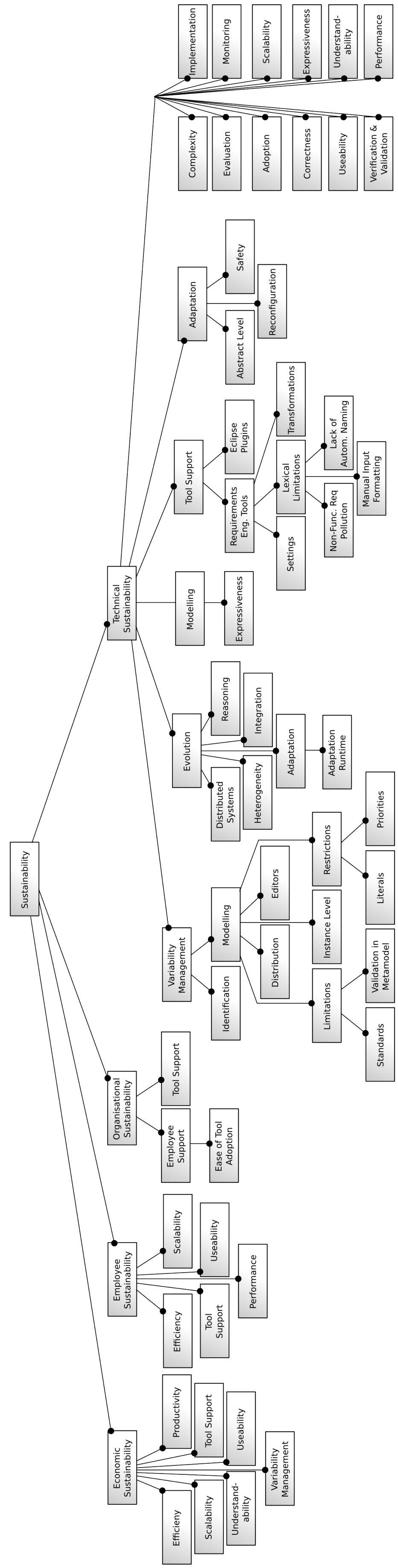

Figure 1: Sustainability Feature Models of the Diva Case Study.