Lean thinking has been successfully employed beyond manufacturing in product design and development. This has led to an increasing interest to utilize it in software engineering (SE). The thesis advances this goal by identifying the key practice of Value Stream Mapping (VSM) that can facilitate organizations to adopt Lean for software development. Two major shortcomings of VSM when applying it in the SE context were identified in this research. Furthermore, solutions were proposed and evaluated to mitigate them. This research was conducted in close collaboration with an industrial partner.

Introduction

Lean thinking has its origins in the Japanese manufacturing industry and can be traced to work done in the 1950s at Toyota [9]. Womack et al. [20] however are credited to coin the term “Lean production” and for bringing widespread attention to Lean thinking [9]. Lean is defined as specifying value from the customer’s standpoint, mapping and streamlining value-creation activities to deliver it (value stream mapping), developing the ability to conduct these activities without interruption and with predictability (achieving flow), where the development is only triggered by a request (pull-based development), and always striving to perform these activities with ever more effectiveness (continuous improvement) [19].

Womack and Jones [19] propose a step-by-step plan for transformation to Lean as the following: (1) Find a change agent, get the necessary knowledge and competence. (2) Find motivation to introduce change (capitalize on a crisis). (3) Perform Value Stream Mapping (VSM). (4) Picking something important, quickly start with removing waste for immediate returns. From the definition, and the transformation steps the central role and contribution of VSM to operationalize Lean thinking and facilitating organizations in transforming their current way of working is evident [17] [19]. Furthermore, many organizations in various domains have attributed substantial success to the VSM based improvements [11].

However, a review of extant literature on Lean in SE shows that VSM lacks the same recognition in SE [13]. In a systematic literature review, Pernstål et al. [13] found that while VSM is a “central practice” in Lean, only two papers report the use of VSM in the SE context out of a total of 38 included papers. It seems that early adopters of Lean in SE are experiencing the common pitfalls as many manufacturing organizations. As Womack and Jones noted that many organizations skip the most critical step of performing VSM itself and instead rush to eliminate waste from the process [17]. A similar trend is seen in the SE context [13] where only a few studies have reported the use of VSM and the focus has been on applying individual Lean practices and investigating various types of waste [16] in software development. Such well-intended endeavors lead to sub-optimization, as only isolated parts of the overall value stream are improved. Thus, the benefits of Lean transformation in the form of improved quality, reduced cost and shorter time-to-market do not reach the end customer.

VSM is a Lean practice that maps the current product development process (current state map), identifies value adding and non-value adding activities and steps, and helps to create an action plan for achieving an improved future state of the process (future state map) [11] [17] [11] [16] [18]. VSM facilitates to disseminate the current understanding of customer value throughout the entire development organization and helps attain alignment with it.

Figure 1 depicts how VSM can systematically operationalize the Lean principles by using them first to guide the analysis of the current value stream, then in identification of waste and lastly in identifying which improvements to implement.
Summary

From key literature on Lean [19] [17] and from an application of VSM in an industrial software development context [10], the following two shortcomings in current adaptations of VSM to the SE context were identified. First, the notation used only provides a snapshot of the system and fails to capture the dynamic aspects of the underlying processes. This leads to a simplistic analysis to identify bottlenecks. Also, the improvement actions and the target value maps are assessed based on idealistic assertions. These limitations reduce the confidence in the improvement actions identified in VSM, which implies that it is less likely that such improvement actions will be implemented. Second, the current VSM method and notation is unable to capture and represent the myriad of significant information flows, which in software development go beyond just the schedule information about a software artifact’s flow through the various phases of a development process.

The thesis attempts to address these shortcomings and emphasizes the use of VSM in the context of Lean software development as a practice that connects existing work on conceptual (e.g., principles of Lean) and tactical levels (concrete practices and processes). For the first shortcoming of VSM, we have proposed the use of Software Process Simulation Modeling (SPSM) as a solution. To achieve this improvement, the usefulness of SPSM was evaluated, and guidelines to perform simulation-based studies in industry were consolidated. This knowledge was used to support VSM with SPSM. To overcome the second shortcoming of VSM, alternatives for capturing rich information flows in software development were explored and a suitable approach was identified to support VSM.

Through three literature reviews, one systematic literature review (SLR), four industrial case studies, and a case study in an academic context that were conducted as part of this research, the thesis makes the following contributions:

1. Recognizes the central role of VSM in operationalization of Lean in the SE context and improves the existing guidelines for conducting VSM.
2. Determined the usefulness of SPSM to support VSM in artifact flow analysis and when reasoning about changing the process.
3. Determined the utility of FLOW to support VSM to capture, analyze and improve information flows in software development.
4. Determined the usefulness of SPSM in applied settings.
5. Consolidated the guidelines to apply SPSM in industry.
6. Improvement in the guidelines for conducting systematic literature studies by providing means to systematically perform and document study selection related decisions.

Table 1 provides an overview of the publications that contributed in this thesis. The following is a brief summary of the contribution and connection between various chapters: Chapters 2 and 3 attempt
to identify evidence of the usefulness of software process simulation for software engineering. Chapter 2 reports the design and results of a systematic literature review of industrial studies on software process simulation, while Chapter 3 is a primary study where the impact of software process simulation was investigated in an academic setting. Using a literature review, we identified and consolidated the process to conduct a simulation-based study. This process, its application in the case company and the lessons learned are reported in Chapter 4. A case study, undertaken to understand the testing process of our industrial partner to support the simulation based study, is also used in this chapter. Chapter 5 reports the lessons learned from other disciplines that have combined value stream mapping with simulation. It also reports a framework and its evaluation in two industrial cases. Chapter 6 reports the proposal to support VSM with FLOW and its evaluation in a large-scale product development case. Chapter 7 reports a literature review to identify strategies to reduce bias and resolve disagreements between reviewers in secondary studies (systematic mapping studies and reviews). This chapter also reports the evaluation of identified strategies by utilizing these in a systematic literature review reported in Chapter 2. This work has been used to support the secondary studies conducted in this thesis.

Results and conclusions

Little evidence to substantiate the claims of the usefulness of SPSM was found. Hence, prior to combining it with VSM, we consolidated the guidelines to conduct an SPSM based study and evaluated the use of SPSM in academic and industrial contexts. In education, it was found to be a useful complement to other teaching methods, and in the industry, it triggered useful discussions and was used to challenge practitioners’ perceptions about the impact of existing challenges and proposed improvements. The combination of VSM with FLOW (a method and notation to capture information flows, since existing VSM adaptations for SE are insufficient for this purpose) was successful in identifying challenges and improvements related to information needs in the process. Both proposals to support VSM with simulation and FLOW led to identification of waste and improvements (which would not have been possible with conventional VSM), generated more insightful discussions and resulted in more realistic improvements.

This thesis characterizes the context and shows how SPSM was beneficial both in the industrial and academic context. FLOW was found to be a scalable, lightweight supplement to strengthen the information flow analysis in VSM. Overall, the thesis has facilitated adoption of Lean thinking in the SE context by supplementing the existing guidelines for conducting VSM. Through successful industrial application, positive evaluation and uptake, the thesis provides evidence of the usefulness and scalability of the improved VSM (including support for simulation and richer information flow modeling) in practice.

References


